

# ΙΟΝΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

## ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ



### Πτυχιακή Εργασία

Εκπαίδευση και Ανάλυση μοντέλων βαθιάς μάθησης για  
αναγνώριση κορονοϊού μέσα από αξονικές ακτινογραφίες με  
την χρήση Transfer learning

**Υποψήφιος:** Μάριος Κωνσταντίνου, Π2015002

**Επιβλέπων:** Θεμιστοκλής Έξαρχος, *Επίκουρος Καθηγητής*

# Εκπαίδευση και Ανάλυση μοντέλων βαθιάς μάθησης για αναγνώριση κορονοϊού μέσα από αξονικές ακτινογραφίες με την χρήση Transfer learning

Η πτυχιακή εργασία εγκρίθηκε από:

## **Επιβλέπων**

- Θεμιστοκλής Έξαρχος, Επίκουρος Καθηγητής Τμήμα Πληροφορικής, Ιόνιο Πανεπιστήμιο

## **Διμελής Επιτροπή**

- Αριστείδης Βραχάτης, Επίκουρος Καθηγητής Τμήμα Πληροφορικής, Ιόνιο Πανεπιστήμιο
- Παναγιώτης Βλάμος, Καθηγητής Τμήμα Πληροφορικής, Ιόνιο Πανεπιστήμιο

## Υπεύθυνη Δήλωση

Με το παρόν κείμενο βεβαιώνω ότι ο κάτωθι υπογράφων είμαι συγγραφέας της παρούσης πτυχιακής εργασίας, η οποία εκπονήθηκε στο πλαίσιο των απαιτήσεων του προγράμματος σπουδών του Τμήματος Πληροφορικής του Ιονίου Πανεπιστημίου και η οποία παραδόθηκε, μετά από έγκριση του επιβλέποντας καθηγητή μου, σε έντυπη και ψηφιακή μορφή στη Γραμματεία του Τμήματος. Επίσης, δηλώνω πως κάθε πηγή που χρησιμοποίησα (βιβλιογραφία, αρθρογραφία, δικτυογραφία), για την υποστήριξη των υποθέσεων της μελέτης και της ερευνάς μου, είναι πλήρως συμβατή με τα ακολουθούμενα επιστημονικά πρότυπα και, επιπλέον, αναφέρεται ρητά, υπό μορφή αναφοράς - παραπομπής, σε όλο το φάσμα κειμένων της παρούσης εργασίας.

Επώνυμο	Κωνσταντίνου
Όνομα	Μάριος
Αριθμός μητρώου	Π2015002
Ημερομηνία	30/09/2022
Υπογραφή	

## 1. Περίληψη

Ο κορονοϊός από τον Δεκέμβριο του 2019 μέχρι τώρα επηρέασε σημαντικά εκατομμύρια ανθρώπους. Δεδομένου την επίδραση που έχει η ασθένεια στα αναπνευστικά όργανα και στον πνευμονολογικό ιστό, η απεικόνιση του αναπνευστικού συστήματος με ακτινογραφίες θώρακος είναι αναγκαία για την παρακολούθηση και έλεγχο της ασθένειας. Υπάρχει ένα πλήθος από έρευνες όπου γίνεται η χρήση μοντέλων βαθιάς μάθησης για την αυτοματοποιημένη διάγνωση κορονοϊού. Παρόλο που τα αποτελέσματα των εν λόγω ερευνών δείχνουν θετικά, ο αριθμός αξονικών ακτινογραφιών κορονοϊού είναι περιορισμένος με αποτέλεσμα να μη γίνεται αξιολόγηση των μοντέλων σε ξεχωριστά δεδομένα. Στην εν λόγω πτυχιακή εργασία αξιολογήθηκαν και αναλύθηκαν 5 μοντέλα βαθιάς μάθησης με σκοπό την αναγνώριση κορονοϊού σε αξονικές ακτινογραφίες. Χρησιμοποιήθηκε το μεγαλύτερο για την ώρα σετ δεδομένων που είναι διαθέσιμο προς το κοινό το οποίο είναι κατακερματισμένο σε *train set* για την εκπαίδευση, *validation set* για την αξιολόγηση της εκπαίδευσης και τέλος ένα *test set* για την τελική αξιολόγηση των μοντέλων σε άγνωστα δεδομένα. Στην εν λόγω πτυχιακή εκπαίδευτηκαν τα μοντέλα *ResNet 50*, *ResNet101*, *DenseNet121*, *DenseNet169* και *InceptionV3* με τη χρήση *Transfer Learning*. Και τα 5 μοντέλα κατάφεραν να δείξουν ακρίβεια 93% και άνω, όπου το *ResNet101* απόδωσε καλύτερα από όλα. Συγκεκριμένα η ακρίβεια του ανήλθε στο **95.7%**, η απώλεια **0.13**, το *Precision* **95.7%** και τέλος το *Recall* **95.6%**.

**Λέξεις κλειδιά:** Βαθιά μηχανική μάθηση, κορονοϊός, ResNet50, ResNet101, DenseNet121, DenseNet169, InceptionV3, Transfer Learning, ακτινογραφίες θώρακα

## 2. Abstract

Coronavirus disease since December 2019 has significantly affected millions of people. Given the effect this disease has on the pulmonary systems of humans, chest radiographic imaging is now a necessity for monitoring the disease and preventing further deaths. Multiple studies showed that Deep Learning models can achieve great results in automatic diagnosis of COVID-19. Although, the number of available chest radiographs regarding COVID-19 is limited, hence these models have not been evaluated on data that's new and unknown to the model. Regarding this thesis, 5 deep learning models were analyzed and evaluated with the aim of the identifying COVID-19 from chest X-Ray images. The dataset that was used is the largest one available publicly by the time of writing this thesis. Furthermore, the dataset is split into a train set for training each model, then there's the validation set for validating the performance during training and then there's the test set for evaluation on data that was not processed by each model. The models that were used are *ResNet50*, *ResNet101*, *DenseNet121*, *DenseNet169* and *InceptionV3* using *Transfer Learning*. All models were able to achieve accuracy of **93%** and above. *ResNet101* was the best performing model, whose accuracy reached **95.7%**, the loss was at **0.13**, *Precision* reached **95.7%** and lastly *Recall* reached **95.6%**.

**Key words:** Deep Learning, COVID-19, ResNet50, ResNet101, DenseNet121, DenseNet169, InceptionV3, Transfer Learning, Chest X-Rays

# Περιεχόμενα

1. Περίληψη .....	3
2. Abstract .....	4
3. Εισαγωγή .....	11
3.1. Κορονοϊός και εργαλεία διάγνωσης .....	11
3.2. Χρήση Ιατρικής απεικόνισης.....	12
3.3. Τεχνητή νοημοσύνη στην Ιατρική.....	14
3.3.1. Τεχνητή νοημοσύνη για την διάγνωση αναπνευστικών νόσων .....	14
3.4. Σκοπός πτυχιακής εργασίας .....	15
4. Βαθιά μάθηση και μηχανική Όραση.....	16
4.1. Εισαγωγή.....	16
4.2. Η αρχή της βαθιάς μηχανικής μάθησης .....	17
4.3. Μηχανική Όραση και Convolutional Neural Networks .....	18
4.3.1. Χρήση Μηχανικής Όρασης στην Ιατρική .....	20
4.3.2. Χρήση ακτινογραφιών σε συνδυασμό με μηχανική όραση .....	23
5. Βιβλιογραφική ανασκόπηση .....	23
6. Δεδομένα.....	33
6.1. COVID-19 CXR dataset .....	34
6.2. RSNA CXR dataset .....	35
6.3. Chest X-Ray Pneumonia dataset .....	35
6.4. PadChest dataset .....	35
6.5. Montgomery και Shenzhen CXR lung masks datasets .....	35
6.6. QaTa-Cov19 CXR infection mask dataset .....	36
7. Μεθοδολογία.....	39
7.1. Αρχιτεκτονικές και μοντέλα βαθιάς μάθησης .....	39
7.1.1. ResNet.....	39
7.1.2. DenseNet.....	40
7.1.3. Inception Network .....	41
7.2. Εργαλεία .....	42
7.3. Βιβλιοθήκες.....	43
7.3.1. NumPy.....	43

7.3.2. Scikit-Learn.....	43
7.3.3. Pandas.....	43
7.3.4. Matplotlib .....	44
7.3.5. Seaborn.....	44
7.3.6. Jupyter Notebook.....	44
7.3.7. Tensorflow .....	44
7.3.8. Keras.....	45
7.4. Προετοιμασία και προ-επεξεργασία δεδομένων.....	45
7.4.1. Προετοιμασία δεδομένων .....	45
7.4.2. Προ-επεξεργασία δεδομένων – <i>Data Augmentation</i> .....	46
7.5. Προσδιορισμός μοντέλου και <i>Transfer Learning</i> .....	48
7.5.1. Transfer Learning .....	48
7.5.2. Προσδιορισμός μοντέλου / Model Definition .....	51
7.5.3. Μετρήσεις αξιολόγησης κατά την εκπαίδευση.....	52
7.5.4. Επανακλήσεις ( <i>Callbacks</i> ) .....	53
7.6. Αρχική εκπαίδευση μοντέλου.....	54
7.6.1. Fine-tuning .....	55
8. Αποτελέσματα.....	56
8.1. ResNet50.....	56
8.1.1. Αποτελέσματα εκπαίδευσης .....	57
8.1.2. Αποτελέσματα της τελικής αξιολόγησης του <i>ResNet50</i> .....	59
8.2. ResNet101.....	60
8.2.1. Αποτελέσματα εκπαίδευσης του <i>ResNet101</i> .....	60
8.2.2. Αποτελέσματα της τελικής αξιολόγησης του <i>ResNet101</i> .....	62
8.3. DenseNet121 .....	63
8.3.1. Αποτελέσματα εκπαίδευσης του <i>DenseNet121</i> .....	64
8.3.2. Αποτελέσματα της τελικής αξιολόγησης του <i>DenseNet121</i> .....	66
8.4. DenseNet 169.....	67
8.4.1. Αποτελέσματα εκπαίδευσης .....	67
8.4.2. Αποτελέσματα της τελικής αξιολόγησης του <i>DenseNet169</i> .....	69
8.5. Inception-V3.....	70
8.5.1. Αποτελέσματα εκπαίδευσης .....	71
8.5.2. Αποτελέσματα της τελικής αξιολόγησης του <i>InceptionV3</i> .....	73

9. Συζήτηση.....	74
9.1. Αντιπαράβολή αποτελεσμάτων.....	74
9.2. Σύγκριση αποτελεσμάτων .....	75
9.3. Περιορισμοί της βαθιάς μηχανικής μάθησης.....	76
9.4. Προβλήματα στην εν λόγω πτυχιακή εργασία .....	76
9.5. Επίλυση προβλημάτων .....	77
9.6. Μελλοντικές προτάσεις .....	78
10. Συμπεράσματα .....	79
11. Βιβλιογραφία .....	80

## Λίστα Πινάκων

Πίνακας 3-1: Σύνοψη χαρακτηριστικών των ασθενών (Ai et al., 2020).....	13
Πίνακας 5-1: Τα περιεχόμενα του κάθε σετ δεδομένων (Narin et al., 2021).....	27
Πίνακας 5-2: Ο κατακερματισμός του σετ δεδομένων σε training, validation και test (Afifi et al., 2021).29	
Πίνακας 5-3: Η απόδοση του κάθε μοντέλου χρησιμοποιώντας Micro-average (Afifi et al., 2021).....	30
Πίνακας 5-4: Απόδοση του COVID-Net συγκριτικά με τις αρχιτεκτονικές VGG-19 και ResNet-50 (L. Wang et al., 2020).....	32
Πίνακας 5-5: Sensitivity του COVID-Net συγκριτικά με τις αρχιτεκτονικές VGG-19 και ResNet-50 (L. Wang et al., 2020).....	32
Πίνακας 5-6: PPV του COVID-Net συγκριτικά με τις αρχιτεκτονικές VGG-19 και ResNet-50 (L. Wang et al., 2020). .....	32
Πίνακας 6-1: Η δομή του σετ δεδομένων COVID-QU. ....	34
Πίνακας 6-2: Ο κατακερματισμός του σετ δεδομένων της εν λόγω πτυχιακής εργασίας. ....	38
Πίνακας 7-1: Παράδειγμα για υπολογισμό μετρήσεων ενός multi-class classification προβλήματος. Ορίζοντα είναι οι τιμές που πρόβλεψε το μοντέλο και κάθετα είναι οι πραγματικές τιμές.....	53
Πίνακας 7-2: Συνοπτικά ο αριθμός εκπαιδευσιμων παραμέτρων του κάθε μοντέλου αφού “ξεπάγωσαν” τα ανάλογα μοντέλα. ....	55
Πίνακας 7-3: Οι τιμές των παραμέτρων που χρησιμοποιήθηκαν για όλα τα μοντέλα. ....	55
Πίνακας 8-1: Ο αριθμός εκπαιδευόμενων και μη παραμέτρων σε κάθε φάση του ResNet50.....	56
Πίνακας 8-2: Πίνακας τελικής αξιολόγησης του μοντέλου ResNet50. ....	59
Πίνακας 8-3: Ο αριθμός εκπαιδευόμενων και μη παραμέτρων σε κάθε φάση του ResNet101.....	60
Πίνακας 8-4: Πίνακας τελικής αξιολόγησης του μοντέλου ResNet101. ....	62



Πίνακας 8-5: Ο αριθμός εκπαιδευόμενων και μη παραμέτρων σε κάθε φάση του DenseNet121.....	63
Πίνακας 8-6: Πίνακας τελικής αξιολόγησης του μοντέλου DenseNet121. ....	66
Πίνακας 8-7: Ο αριθμός εκπαιδευόμενων και μη παραμέτρων σε κάθε φάση του DenseNet169.....	67
Πίνακας 8-8: Πίνακας τελικής αξιολόγησης του μοντέλου DenseNet169. ....	69
Πίνακας 8-9: Ο αριθμός εκπαιδευόμενων και μη παραμέτρων σε κάθε φάση του InceptionV3. ....	70
Πίνακας 8-10: Ο πίνακας τελικής αξιολόγησης του μοντέλου InceptionV3.....	73
Πίνακας 9-1: Σύνοψη αποτελεσμάτων της τελικής αξιολόγησης του κάθε μοντέλου. ....	74

## Λίστα Εικόνων

Εικόνα 3-1: Συνολικοί θάνατοι παγκοσμίως μέχρι 28 Ιουνίου, 2022 (Coronavirus Graphs, n.d.).....	11
Εικόνα 3-2: Μερικές εικόνες με τις κατηγορίες τους, που θα χρησιμοποιηθούν στην εν λόγω πτυχιακή εργασία.....	15
Εικόνα 4-1: Διαφορές μεταξύ των τριών επιστημών (Aggarwal et al., 2022).....	16
Εικόνα 4-2: Παράδειγμα νευρικών ομάδων (Bain, 1873). ....	17
Εικόνα 4-3: Η δομή ενός Perceptron του τότε (Rosenblatt, 1958) (β) Ένας perceptron του σήμερα (Wang & Raj, 2017).....	18
Εικόνα 4-4: Συνοπτικά η ροή της πληροφορίας που περνά από τον οπτικό φλοιό του ανθρώπινου οπτικού συστήματος. (Wang & Raj, 2017). ....	18
Εικόνα 4-5: Ένα απλό Convolutional Operation ενός δισδιάστατου πίνακα (H. Wang & Raj, 2017).....	19
Εικόνα 4-6: Παράδειγμα ενός Subsampling Layer (Yingge et al., 2020).....	19
Εικόνα 4-7: Αρχιτεκτονική LeNet (Lecun et al., 1998).....	19
Εικόνα 4-8: Δύο μοντέλα (a) και (b). Το πρώτο (a) παίρνει ως είσοδο εικόνες και κείμενο για να κάνει μία διάγνωση. Το (b) είναι ένα Generative Model για την αναπαραγωγή εικόνων. (Esteva et al., 2021) .....	20
Εικόνα 4-9: Ένα μοντέλο βαθιάς μάθησης ταξινομεί ταυτόχρονα 15 διαφορετικά ηχοκαρδιογραφήματα (Madani et al., 2018).....	21
Εικόνα 4-10: Η δομή του συστήματος Deep Profiler (Lou et al., 2019). ....	22
Εικόνα 5-1: Η ολοκληρωμένη δομή του συστήματος (Toğaçar et al., 2020). ....	23
Εικόνα 5-2: Το σύστημα που δημιούργησαν οι (Degerli et al., 2021) για την αναγνώριση κορονοϊού.....	25
Εικόνα 5-4: Το ολοκληρωμένο σύστημα της προκυμμένης έρευνας (Keidar et al., 2021). ....	26
Εικόνα 5-5: Η δομή του συστήματος (Afifi et al., 2021).....	29
Εικόνα 5-6: Η αρχιτεκτονική COVID-Net (L. Wang et al., 2020). ....	31
Εικόνα 6-1: Δείγμα από το σετ δεδομένων που θα χρησιμοποιηθεί στην προκυμμένη πτυχιακή εργασία.34	

Εικόνα 6-2: Τα αποτελέσματα της αξιολόγησης των μοντέλων όσο αφορά την παραγωγή των масκών (Tahir et al., 2021).	37
Εικόνα 7-1: Συνοπτικά η μεθοδολογία της εν λόγω πτυχιακής εργασίας.	39
Εικόνα 7-2: Ένα Residual Block της αρχιτεκτονικής ResNet (He et al., 2015).	40
Εικόνα 7-3: Ένα DenseBlock με 5 layers (Huang et al., 2018).	40
Εικόνα 7-4: Ο αρχικός σχεδιασμός του Inception Module (Szegedy et al., 2014).	41
Εικόνα 7-5: Το διάγραμμα της Inception-v3 αρχιτεκτονικής (Advanced Guide to Inception v3   Cloud TPU, n.d.).	41
Εικόνα 7-6: Παράδειγμα εφαρμογής Data Augmentation 9 φορές σε 1 εικόνα του σετ δεδομένων της προκυμμένης πτυχιακής εργασίας	47
Εικόνα 7-7: Παράδειγμα εκμάθησης ενός μοντέλου σε κάθε επίπεδο. Οι εικόνες πάρθηκαν από το βιβλίο Deep Learning with python (Chollet, 2018).	49
Εικόνα 7-8: Συνοπτικά τα βήματα για την υλοποίηση TL (Chollet, 2018).	50
Εικόνα 7-9: Υλοποίηση fine-tuning όπου μερικά επίπεδα του μοντέλου "ξεπαγώνουν" και εκπαιδεύονται. Στη φάση Α, τα επίπεδα του μοντέλου είναι "παγωμένα" και εκπαιδεύεται μόνο ο καινούριος ταξινομητής. Στη φάση Β, "ξεπαγώνουν" μερικά επίπεδα και το μοντέλο ξανά εκπαιδεύεται.	51
Εικόνα 8-1: Η ακρίβεια του μοντέλου ResNet50.	57
Εικόνα 8-2: Η απώλεια του μοντέλου κατά την εκπαίδευση ResNet50.	57
Εικόνα 8-3: Το Precision του μοντέλου ResNet50.	58
Εικόνα 8-4: Το Recall του μοντέλου ResNet50.	58
Εικόνα 8-5: Το confusion matrix του μοντέλου ResNet50.	59
Εικόνα 8-6: Η ακρίβεια του μοντέλου ResNet101.	60
Εικόνα 8-7: Η απώλεια του μοντέλου ResNet101.	61
Εικόνα 8-8: Το Precision του μοντέλου ResNet101.	61
Εικόνα 8-9: Το Recall του μοντέλου ResNet101.	62
Εικόνα 8-10: Το confusion matrix του μοντέλου ResNet101.	63
Εικόνα 8-11: Η ακρίβεια του μοντέλου DenseNet121.	64
Εικόνα 8-12: Η απώλεια του μοντέλου DenseNet121.	64
Εικόνα 8-13: Το Precision του μοντέλου DenseNet121.	65
Εικόνα 8-14: Το Recall του μοντέλου DenseNet121.	65
Εικόνα 8-15: Το confusion matrix του μοντέλου DenseNet121.	66
Εικόνα 8-16: Η ακρίβεια του μοντέλου DenseNet169.	67
Εικόνα 8-17: Η απώλεια του μοντέλου DenseNet169.	68
Εικόνα 8-18: Το Precision του μοντέλου DenseNet169.	68

Εικόνα 8-19: Το Recall του μοντέλου DenseNet169.....	69
Εικόνα 8-20: Το confusion matrix του μοντέλου DenseNet169.....	70
Εικόνα 8-21: Η ακρίβεια του μοντέλου InceptionV3. ....	71
Εικόνα 8-22: Η απώλεια του μοντέλου InceptionV3.....	71
Εικόνα 8-23: Το Precision του μοντέλου InceptionV3.....	72
Εικόνα 8-24: Το Recall του μοντέλου InceptionV3. ....	72
Εικόνα 8-25: Το confusion matrix του μοντέλου InceptionV3. ....	73

## Συντομογραφίες

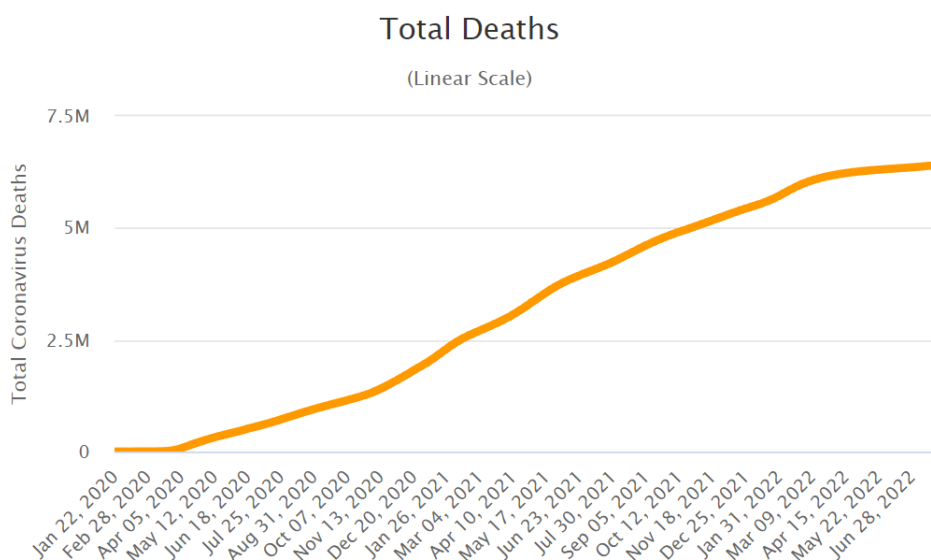
Πίνακας Συντομογραφιών	
DL	Deep Learning - Βαθιά Μάθηση
ML	Machine Learning - Μηχανική Μάθηση
CNN	Convolutional Neural Networks
SM	Systems Medicine
SVM	Support Vector Machine
SMO	Sequential minimal optimization
Area under the curve	AUC
Receiver operating characteristics	ROC
Transfer Learning	TL
True Positives	TP
True Negatives	TN
False Positives	FP
False Negatives	FN
Learning Rate	lr

### 3. Εισαγωγή

#### 3.1. Κορονοϊός και εργαλεία διάγνωσης

Τον Δεκέμβριο του 2019 στο Wuhan της Κίνας, διαγνώστηκαν ασθενείς με πνευμονία, άγνωστης προέλευσης. Μετά από ανάλυση του αναπνευστικού συστήματος των ασθενών, ένα νέο είδος Κορονοϊού ανακαλύφθηκε (*Emerging 2019 Novel Coronavirus (2019-NCoV) Pneumonia*, n.d.). Μέσα σε μόλις 3 μήνες (Ιανουάριος 2020 – Μάρτιος 2020), τα κρούσματα του κορονοϊού έφτασαν στο μισό εκατομμύριο (*Coronavirus Graphs*, n.d.) και εξαιτίας αυτής της ραγδαίας αύξησης, ο οργανισμός WHO, χαρακτήρισε τον COVID-19 ως πανδημία στις 11 Μαρτίου 2020 (*WHO Director-General's Opening Remarks at the Media Briefing on COVID-19 - 11 March 2020*, n.d.).

Μερικά συνηθισμένα συμπτώματα που μπορεί να παρουσιάσουν ασθενείς με κορονοϊό είναι πυρετός, βήχας, κόπωση, πονοκέφαλο, ζαλάδα, δύσπνοια και πτύελο (ALIMOHAMADI et al., 2020; *Emerging 2019 Novel Coronavirus (2019-NCoV) Pneumonia*, n.d.), αυτά τα συμπτώματα, συνάδουν με το γενικό σύμπτωμα το οποίο είναι ή πνευμονία (ALIMOHAMADI et al., 2020). Παράλληλα, παρατηρήθηκε πως ασθενείς που πάσχουν από ιογενή πνευμονία κορονοϊού υπέστησαν ζημιές στο αναπνευστικό τους σύστημα και πιο συγκεκριμένα, εντοπίστηκαν κακώσεις στους χαμηλότερους λοβούς και στους δύο πνεύμονες (*Emerging 2019 Novel Coronavirus (2019-NCoV) Pneumonia*, n.d.). Μέχρι τώρα, οι θάνατοι που οφείλονται στον COVID-19 ανέρχονται στους 6,364,409 παγκοσμίως (*Coronavirus Graphs*, n.d.) (βλ. *Εικόνα 3-1*).



**Εικόνα 3-1: Συνολικοί θάνατοι παγκοσμίως μέχρι 28 Ιουνίου, 2022 (*Coronavirus Graphs*, n.d.)**

Με νέα κύματα και νέες μεταλλάξεις κορονοϊού να εμφανίζονται συνεχώς, η αναγκαιότητα για γρήγορη και ακριβή διάγνωση είναι πιο σημαντική από ποτέ. Πέρα από τα εργαλεία διάγνωσης που θα σχολιαστούν παρακάτω, έγιναν έρευνες και πειράματα πάνω στη χρήση τεχνητής νοημοσύνης σε συνδυασμό με ακτινολογικές φωτογραφίες για τη διάγνωση ιογενούς πνευμονίας κορονοϊού αλλά και άλλων πνευμονολογιών ασθενειών.

Υπάρχουν διάφορα εργαλεία για τη διάγνωση του Κορονοϊού. Μερικά από αυτά είναι τα RT-PCR τεστ, self-test, αξονικές τομογραφίες και ακτινογραφίες. Η πιο διαδεδομένη μέθοδος μέχρι τώρα είναι τα RT-PCR τεστ, τα οποία θεωρούνται ως η κύρια μέθοδος διάγνωσης (Tahamtan & Ardebili, 2020). Η απόδοση και η ακρίβεια των self-test συγκριτικά με τα RT-PCR test είναι αρκετά μικρότερη και τα αποτελέσματα τους μπορεί να είναι ψευδή θετικά ή ψευδή αρνητικά, κάνοντάς τα ακατάλληλα για ακριβή διάγνωση, ειδικά στην περίπτωση μη-συμπτωματικών ασθενών (Iftner et al., 2022). Παράλληλα, τα RT-PCR test έχουν τα δικά τους μειονεκτήματα, όπως ψευδή θετικά ή ψευδή αρνητικά αποτελέσματα, σε μικρότερο ποσοστό συγκριτικά με τα self-test. Επίσης, οι διάφοροι τρόποι δειγματοληψίας μπορούν να επηρεάσουν την ακρίβεια του τεστ (Y. Wang et al., 2020). Για αυτό τον λόγο, τα αποτελέσματα αυτών των τεστ δεν πρέπει να είναι το μόνο κριτήριο για διάγνωση (Tahamtan & Ardebili, 2020).

### 3.2. Χρήση Ιατρικής απεικόνισης

Όσο αφορά τη χρήση ακτινογραφιών ή αξονικών τομογραφιών, η πλειονότητα των περιστατικών Κορονοϊού, δείχνουν να έχουν παρεμφερή χαρακτηριστικά μεταξύ τους. Όπως αμφοτερόπλευρες ανωμαλίες, περιφερειακό ή οπίσθιο ground-glass opacity (GGO) κυρίως στους κάτω λοβούς στα αρχικά στάδια και πνευμονική εδραίωση στα τελικά στάδια (C. Huang et al., 2020).

Αυτά τα χαρακτηριστικά δεν φαίνονται χωρίς τη χρήση Ιατρικής απεικόνισης και για αυτό τον λόγο η χρήση τους είναι σημαντική για τη διάγνωση και σωστή αγωγή ασθενών κορονοϊού. Βέβαια, ή διάγνωση πνευμονίας μέσω ακτινολογικών φωτογραφιών είναι μία δύσκολη διαδικασία και προαπαιτεί τη διαθεσιμότητα ειδικών ραδιολόγων. Κλινικές μικρής κλίμακας ενδέχεται να μην έχουν διαθέσιμους γιατρούς που να μπορούν να κάνουν μία τέτοια διάγνωση, και παράλληλα η ραγδαία αύξηση των κρουσμάτων μεγαλώνει το φορτίο που έχουν η ήδη διαθέσιμοι γιατροί.

Σε μία μελέτη, έγινε η χρήση αξονικής τομογραφίας (CT Scans) σε συνδυασμό με RT-PCR τεστ για τη διάγνωση Κορονοϊού και πνευμονίας όπου έδειξαν πως ο συνδυασμός αξονικής τομογραφίας με RT-PCR τεστ προσφέρει μεγαλύτερη ακρίβεια στη διάγνωση (Y. Wang et al., 2020).

**Πίνακας 3-1: Σύνοψη χαρακτηριστικών των ασθενών (Ai et al., 2020).**

<b>Table 1: Summary of Patient Characteristics</b>	
Characteristic	Value
No. of patients	1014
Age (y)	
Mean $\pm$ standard deviation*	51 $\pm$ 15 (2–95)
<20	7 (1)
20–39	267 (26)
40–59	409 (40)
$\geq 60$	331 (33)
No. of men	467 (46)
Median time between chest CT and RT-PCR assay (d)*	1 (0–7)
Results of RT-PCR assay	
Positive	601 (59)
Negative	413 (41)
Findings and manifestations of chest CT	
Consistent with viral pneumonia (positive)	888 (88)
Ground-glass opacity	409/888 (46)
Consolidation	447/888 (50)
Reticulation and/or thickened interlobular septa	8/888 (1)
Nodular lesions	24/888 (3)
No CT findings of viral pneumonia	126 (12)

Note.—Except where indicated, data are numbers of patients with percentages in parentheses. RT-PCR = reverse-transcription polymerase chain reaction.  
\* Numbers in parentheses are the range.

Επιπρόσθετα, σε μία άλλη μελέτη έγινε χρήση αξονικής τομογραφίας σε συνδυασμό με RT-PCR τεστ σε 1014 ασθενείς από διάφορες ηλικιακές ομάδες και των δύο φύλων (βλ. Πίνακα 3-1). Οι ερευνητές βρήκαν πως οι αξονικές τομογραφίες έχουν υψηλά επίπεδα ευαισθησίας στη διάγνωση του κορονοϊού. Επίσης, αρνητικά RT-PCR τεστ αλλά θετικά κλινικά ευρήματα στην αξονική τομογραφία υποδηλώνουν την παρουσία κορονοϊού στον ασθενή. Πέρα από αυτό, οι ερευνητές έβγαλαν το συμπέρασμα πως οι αξονικές τομογραφίες θα μπορούσαν να χρησιμοποιηθούν για τη γρήγορη και ακριβή διάγνωση του κορονοϊού, εξαιτίας της μεγάλης ευαισθησίας που προσφέρουν (Ai et al., 2020).

Βέβαια, όσο αφορά τη χρήση ακτινογραφιών ή αξονικών τομογραφιών, οι ακτινογραφίες είναι πιο φθηνές, γρηγορότερες, και είναι μία διαδεδομένη μέθοδος, διαθέσιμη προς το κοινό, όπου το σώμα εκτίθεται σε μικρότερα επίπεδα ραδιενέργειας σε σχέση με τις αξονικές τομογραφίες (Brenner & Hall, 2007). Ακτινογραφίες θώρακα χρησιμοποιούνται ήδη σε μεγάλο βαθμό ως διαγνωστικό εργαλείο ενάντια στον κορονοϊό και έχει δηλωθεί πως υπάρχει μεγάλη προοπτική να χρησιμοποιηθεί ως εργαλείο πρόγνωσης (Shi et al., 2021).

Η τεχνητή νοημοσύνη είναι ένας κλάδος της πληροφορικής, που προσπαθεί να αναπτύξει μηχανές που να μπορούν να εκτελέσουν ενέργειες οι οποίες πέφτουν στο φάσμα της ανθρώπινης νοημοσύνης. Στη σημερινή εποχή, η τεχνητή νοημοσύνη έχει ενεργό ρόλο στην καθημερινότητα του ανθρώπου. Χρησιμοποιείται παντού, όπως σε σπίτια, οικιακές και φορητές συσκευές, αυτοκίνητα, εργοστάσια, και σε πολλά άλλα. Παράλληλα εισήλθε σε πολλούς κλάδους όπως στη βιομηχανία, γεωργία και ιατρική. Τα τελευταία χρόνια έχει δει τεράστια άνοδο (Aggarwal et al., 2022) και πλέον η χρήση της έχει επεκταθεί σε

αρκετά αντικείμενα όπως α) Αυτοματισμός β) μηχανική μάθηση / μηχανική όραση γ) επεξεργασία ομιλίας και γλώσσας δ) ρομποτική και ε) αυτό-οδηγούμενα οχήματα (Tai, 2020).

### 3.3. Τεχνητή νοημοσύνη στην Ιατρική

Πιο συγκεκριμένα, η μηχανική μάθηση / Machine Learning (ML) χρησιμοποιείται σε διάφορους κλάδους της Ιατρικής όπως κατηγοριοποίηση καρδιοαγγειακών παθήσεων (Alghamdi et al., 2020; Hammad et al., 2018), κατηγοριοποίηση διαβητικής αμφιβληστροειδοπάθειας (Khalil et al., 2019) και κατηγοριοποίηση κερατοειδούς μοτίβων (Haggag et al., 2019). Πέρα από αυτό, μοντέλα βαθιά μάθησης μπορούν να χρησιμοποιηθούν για ανάλυση ακτινογραφιών ως εργαλεία διάγνωσης ή ως εργαλεία υποστήριξης αποφάσεων ενάντια στο κορονοϊό (Qaid et al., 2021).

Η βαθιά μηχανική μάθηση (Deep Learning / DL) είναι υποκατηγορία της μηχανικής μάθησης. Στόχος αυτής της επιστήμης είναι η δημιουργία ενός υπολογιστικού συστήματος που προσομοιάζει το ανθρώπινο μυαλό. Η δημιουργία ενός τέτοιου συστήματος προαπαιτεί την κατανόηση των λειτουργιών του δικού μας νοητικού συστήματος (H. Wang & Raj, 2017). Αυτά τα τεχνητά νευρικά δίκτυα μπορούν να κατηγοριοποιηθούν σε διάφορες ομάδες όπως CNNs (Convolutional Neural Networks), RNNs (Recurrent Neural Networks), GANs (Generative Neural Networks), και άλλα.

#### 3.3.1. Τεχνητή νοημοσύνη για την διάγνωση αναπνευστικών νόσων

Γίνονται πολλές προσπάθειες και μελέτες για τη διάγνωση αναπνευστικών νόσων όπως την ιογενή πνευμονία κορονοϊού, με τη χρήση αξονικών φωτογραφιών και ακτινογραφιών σε συνδυασμό με μοντέλα βαθιάς μάθησης / Deep Learning (DL) (Arias-Garzón et al., 2021, p. 19; DeGrave et al., 2021; Jain et al., 2021).

Συγκεκριμένα, σε μία έρευνα, έγινε εκπαίδευση μερικών Deep Convolutional Neural Networks με την χρήση Transfer Learning (TL) για την αυτόματη διάγνωση κορονοϊού μέσω X-Ray φωτογραφιών όπου μερικά μοντέλα είχαν ακρίβεια 99% (Chowdhury et al., 2020). Επίσης, σε μία άλλη έρευνα δημιουργήθηκε ένας αλγόριθμος βαθιάς εκμάθησης (CheXNet) με συνολικά 121 convolutional επίπεδα, όπου κατάφερε σε μερικές περιπτώσεις να αποδώσει καλύτερα από ειδικούς ραδιολόγους (Rajpurkar et al., 2017).

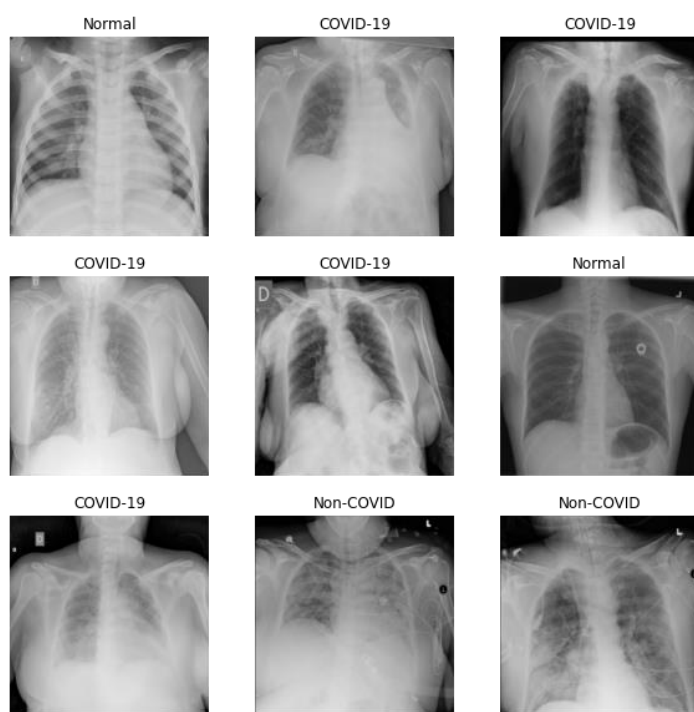
Επιπρόσθετα, επιστήμονες δημιούργησαν ένα αλγόριθμο όπου εφαρμόζει Data Augmentation στις εικόνες, ύστερα διαχωρίζει το κομμάτι των πνευμόνων από τις εικόνες και στη συνέχεια, τροφοδοτούνται οι εικόνες που περιέχουν μόνο τους πνεύμονες σε ένα νευρικό δίκτυο. Το νευρικό δίκτυο είναι υπεύθυνο για δύο δουλειές, α) την ομαδοποίηση παρόμοιων εικόνων με αυτή που τροφοδοτείται και β) την κατηγοριοποίηση

της εικόνας σε Θετικό ή Αρνητικό κορονοϊό. Ο αλγόριθμος κατάφερε να φτάσει σε ακρίβεια 90.3%, specificity 90% και sensitivity 90.5% (Keidar et al., 2021).

Οι τεχνικές Data Augmentation και Transfer Learning θα χρησιμοποιηθούν στην παρούσα εργασία και θα γίνει επεξήγηση τους στο κεφάλαιο *Μεθοδολογία*.

### 3.4. Σκοπός πτυχιακής εργασίας

Σκοπός αυτής της πτυχιακής εργασίας είναι να γίνει εκτίμηση στο αν ένα μοντέλο βαθιάς μάθησης μπορεί να χρησιμοποιηθεί ως εργαλείο υποστήριξης λήψης αποφάσεων για τη διάγνωση ασθενών με κορονοϊό. Σχετικά με αυτό, θα γίνει ανάλυση και σύγκριση μερικών μοντέλων (διαφόρων αρχιτεκτονικών βαθιάς μάθησης) στο πόσο εφικτό είναι η δημιουργία ενός συστήματος αυτόματης ανίχνευσης κορονοϊού μέσω ακτινογραφιών. Τα μοντέλα που θα χρησιμοποιηθούν βασίζονται πάνω στις αρχιτεκτονικές *ResNet50*, *ResNet101*, *DenseNet121*, *DenseNet169*, *Inception V3*.



*Εικόνα 3-2: Μερικές εικόνες με τις κατηγορίες τους, που θα χρησιμοποιηθούν στην εν λόγω πτυχιακή εργασία.*

Τα δεδομένα που θα χρησιμοποιηθούν (βλ. *Εικόνα 3-2*) για την εκπαίδευση, ανάλυση και αξιολόγηση είναι πνευμονολογικές αξονικές φωτογραφίες χωρισμένες σε 3 ομάδες. “Normal”, “Non-COVID” και “COVID-19”.

Η ομάδα “Normal” περιλαμβάνει εικόνες όπου οι πνεύμονες δεν πάσχουν από κάποια πνευμονολογική ασθένεια. Στη συνέχεια η ομάδα “Non-COVID” περιλαμβάνει εικόνες όπου υπάρχει πνευμονολογική ασθένεια αλλά δεν έχει να κάνει με τον κορονοϊό. Τέλος, η ομάδα “COVID-19” περιλαμβάνει εικόνες όπου υπάρχει πνευμονολογική ασθένεια που σχετίζεται με κορονοϊό. Συνολικά το σετ δεδομένων

περιλαμβάνει 33,920 εικόνες. Θα γίνει εκτενής ανάλυση των δεδομένων που χρησιμοποιήθηκαν στο κεφάλαιο *Δεδομένα*.

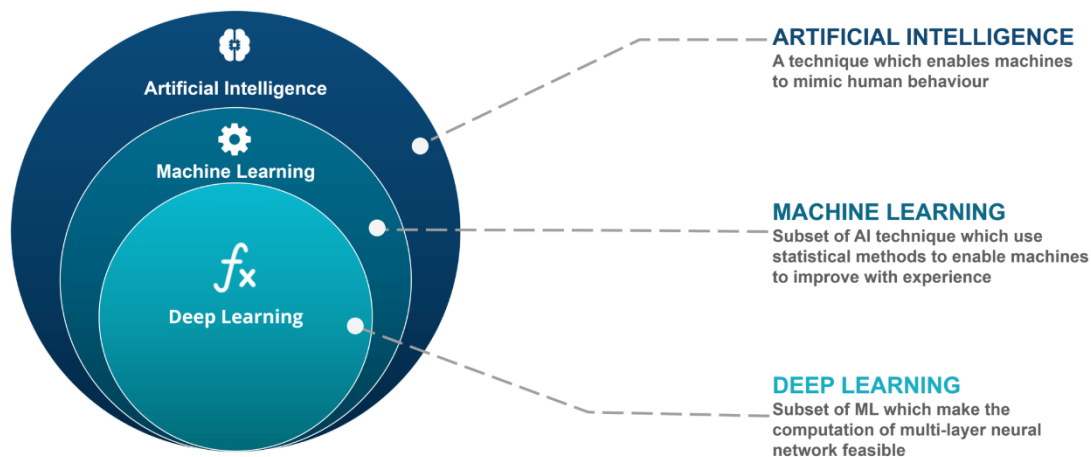


Η ανάλυση θα περιλαμβάνει στατιστικές μετρήσεις όπως *Accuracy*, *Precision*, *Recall* και *F1-Score*. Σχετικά με τις μετρήσεις που αναφέρθηκαν, θα γίνει σχολιασμός για τον χρόνο εκπαίδευσης του κάθε μοντέλου, αλλά και για τον χρόνο πρόβλεψης. Σκοπός είναι η εύρεση ενός μοντέλου που να φτάνει υψηλά επίπεδα απόδοσης, μέσα σε ένα αποδεκτό χρονικό διάστημα. Πιο συγκεκριμένα, ένα μοντέλο θα θεωρείται πως έχει καλή απόδοση, όταν αρχικά α) η ακρίβεια κατά την εκπαίδευση θα ξεπερνάει το 90% β) η ακρίβεια κατά την αξιολόγηση θα παραμένει 90% ή θα αυξάνεται και γ) το μοντέλο δεν θα έχει μεγάλο βαθμό από ψευδή θετικά ή αρνητικά. Δηλαδή, μια φωτογραφία ασθενή με κορονοϊό να κατηγοριοποιείτε από το μοντέλο ως φωτογραφία ασθενή χωρίς κορονοϊό (Ψευδή αρνητικό) και το αντίστροφο όπου ή φωτογραφία ενός ασθενή είναι αρνητική σε κορονοϊό και το μοντέλο την κατηγοριοποιεί ως θετική με κορονοϊό (Ψευδή θετικό).

## 4. Βαθιά μάθηση και μηχανική Όραση

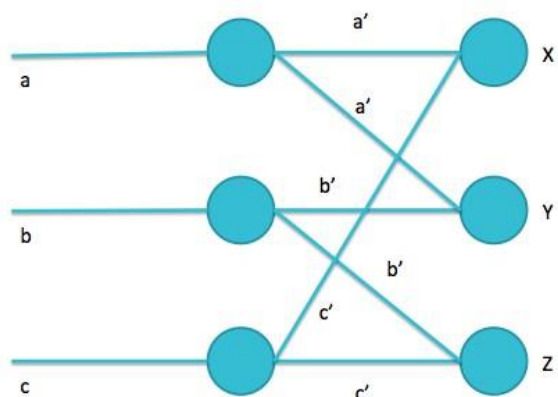
### 4.1. Εισαγωγή

Η Βαθιά Μάθηση είναι τομέας της Μηχανική Μάθησης όπου αυτά τα δύο ανήκουν στο χώρο του Artificial Intelligence (AI) / Τεχνητής Νοημοσύνης. Η βασική ιδέα της Βαθιάς Μάθησης είναι η δημιουργία ενός υπολογιστικού συστήματος που να προσομοιάζει τον τρόπο λειτουργίας του ανθρώπινου μυαλού.



Εικόνα 4-1: Διαφορές μεταξύ των τριών επιστημών (Aggarwal et al., 2022).

## 4.2. Η αρχή της βαθιάς μηχανικής μάθησης



Εικόνα 4-2: Παράδειγμα νευρικών ομάδων (Bain, 1873).

σχέτισε τη θεωρία της συνειρμικής σκέψης με Neural Groupings (βλ. Εικόνα 4-2). Τα Neural Groupings ουσιαστικά απεικονίζουν τη θεωρία της συνειρμικής σκέψης. Για να υπάρξει η σκέψη (**X**) πρέπει πρώτα να ακολουθήσουν οι σκέψεις (**a**), (**c**) και έπειτα (**a'**) μαζί με (**c'**). Δύο απλές σκέψεις οδηγούν σε μία σύνθετη (H. Wang & Raj, 2017).

Στη συνέχεια έχουμε τον κανόνα “Hebbian Learning Rule” και πήρε το όνομα του από τον Donald O. Hebb από το βιβλίο του *The Organization of Behavior* (Hebb, 1949). Ο Hebb έθεσε τον παρακάτω κανόνα:

“Όταν ένας νευράξονας του κυττάρου A είναι αρκετά κοντά στο να διεγείρει ένα κύτταρο B, και το κάνει συνεχώς, τότε δημιουργείται μία μεταβολική αλλαγή σε ένα ή και στα δύο κύτταρα όπου η αποδοτικότητα του A στο να διεγείρει το B, αυξάνεται.”

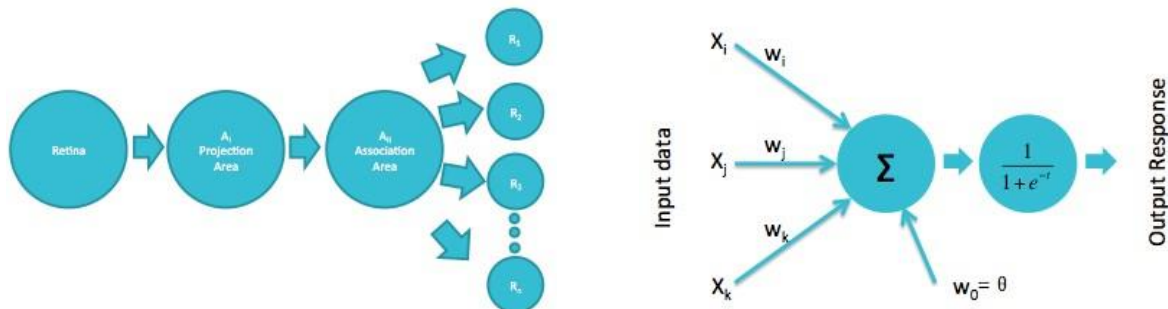
Αυτή η παράγραφος μπορεί να ξαναγραφτεί σε γλώσσα μηχανικής μάθησης ως:

$$\Delta w_i = \eta x_i y$$

Όπου το  $\Delta w_i$  καθορίζει την αλλαγή των βαρών  $w_i$  του νευρώνα  $i$  όπου η είσοδος του σήματος είναι το  $x_i$  και το  $y$  υποδηλώνει τον ρυθμό της εκμάθησης. Με λίγα λόγια, ο κανόνας αυτός λέει πως η ένωση μεταξύ δύο μονάδων πρέπει να ενδυναμώνεται όσο η συχνότητα της συνύπαρξής τους μεγαλώνει. Ο κανόνας Hebbian Learning Rule θεωρείται ως η θεμελιώδης αρχή των νευρικών δικτύων (H. Wang & Raj, 2017).

Παρόλο που ο Donald Hebb θεωρείται ως ο “πατέρας” των νευρικών δικτύων, το πρώτο μοντέλο νευρικού δικτύου δημιουργήθηκε λίγα χρόνια πριν τον κανόνα του Hebb από τον Warren McCulloch και τον Walter Pitts (H. Wang & Raj, 2017). Το μοντέλο λέγεται MCP Neural Model (McCulloch & Pitts, 1943) και

δημιουργήθηκε από ηλεκτρικά κυκλώματα. Στη συνέχεια, μετά την επιτυχία του MCP, ο Frank Rosenblatt παρουσίασε ένα νέο είδος νευρώνα με το όνομα “Perceptron” (Rosenblatt, 1958). Αυτό το σύστημα είχε την ικανότητα να μάθει με βάση την θεωρία της συνειρμικής σκέψης ή “associationism”.

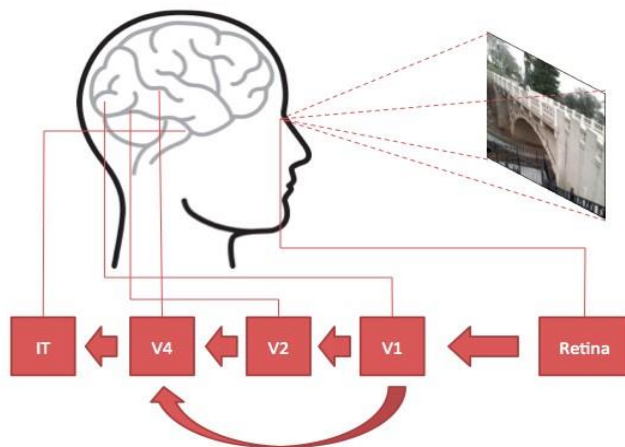


Εικόνα 4-3: Η δομή ενός Perceptron του τότε (Rosenblatt, 1958) (β) Ένας perceptron του σήμερα (Wang & Raj, 2017)

Με την πάροδο του χρόνου δημιουργήθηκαν διάφορα είδη και οικογένειες μοντέλων. Μερικά από αυτά είναι: Hopfield Network (Hopfield, 1982), Boltzmann Machine (Ackley et al., 1985), Restricted Boltzmann Machine (Smolensky, 1986), Deep Belief Nets (Hinton et al., 2006) και Deep Boltzmann Machine (Salakhutdinov & Hinton, 2009).

### 4.3. Μηχανική Όραση και Convolutional Neural Networks

Τα Convolutional Neural Networks είναι μία άλλη οικογένεια μοντέλων η οποία δημιουργήθηκε με βάση την γνώση που έχουμε από τον ανθρώπινο οπτικό φλοιό (H. Wang & Raj, 2017). Μερικά δομικά στοιχεία των CNN είναι το *Convolutional Layer* και το *Subsampling Layer*. Τα Convolutional Layers είναι ουσιαστικά μία μαθηματική διαδικασία που απαρτίζεται από πρόσθεση και πολλαπλασιασμό.



Εικόνα 4-4: Συνοπτικά η ροή της πληροφορίας που περνά από τον οπτικό φλοιό του ανθρώπινου οπτικού συστήματος. (Wang & Raj, 2017).

22	15	1	3	60
42	5	38	39	7
28	9	4	66	79
0	2	25	12	17
9	14	2	51	3

 $\ast$ 

0	0	1
0	0	0
1	0	0

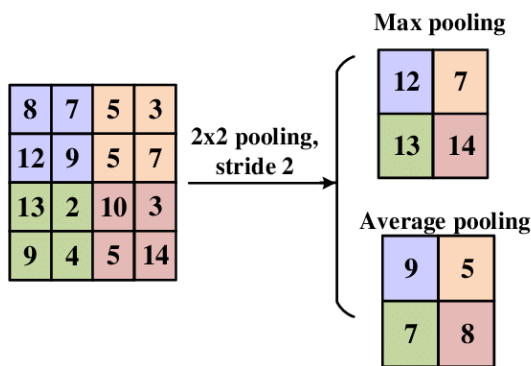
 $=$ 

29	12	64
38	41	32
13	80	81

Εικόνα 4-5: Ένα απλό Convolutional Operation ενός δισδιάστατου πίνακα (H. Wang & Raj, 2017)

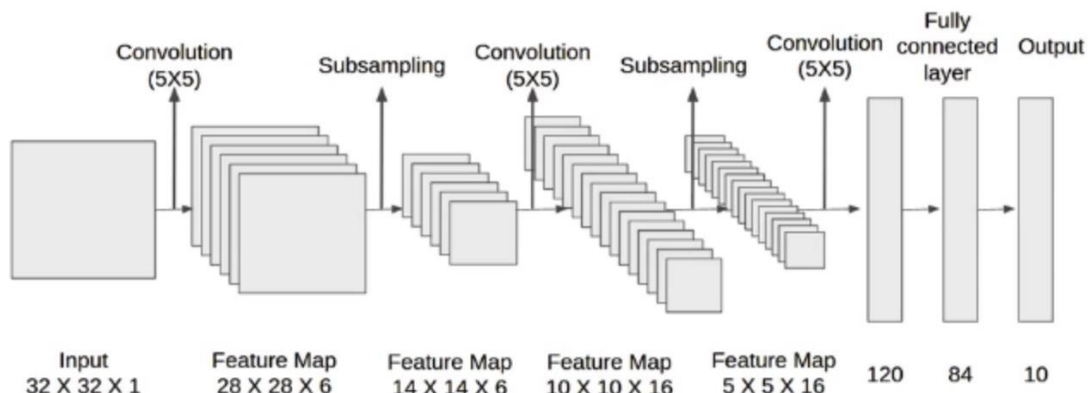
Στην Εικόνα 4-5 φαίνεται: (α) Ο αριστερός πίνακας είναι η είσοδος, (β) ο μεσαίος λέγεται “kernel” και (γ) ο δεξιός πίνακας είναι το αποτέλεσμα του Convolution Operation μεταξύ αυτών των δύο πινάκων.

Το Subsampling Layer (βλ. Εικόνα 4-6) έχει ως είσοδο ένα πίνακα, και στη συνέχεια κάνει δειγματοληψία αυτού του πίνακα παίρνοντας το μέγιστο (max-pooling) ή τον μέσο όρο (average-pooling), ανάλογα ποια μέθοδο θα επιλεγεί (Lee et al., 2009).



Εικόνα 4-6: Παράδειγμα ενός Subsampling Layer (Yingge et al., 2020).

Με την πάροδο του χρόνου δημιουργήθηκαν διάφορες αρχιτεκτονικές οι οποίες είναι βασισμένες σε CNNs. Μία αρχιτεκτονική είναι ουσιαστικά τα δομικά στοιχεία που αναφέρθηκαν παραπάνω, στοιβαγμένα μεταξύ τους συναρμολογώντας ένα Convolutional Neural Network. Ένα κλασικό παράδειγμα είναι η αρχιτεκτονική LeNet (Lecun et al., 1998) όπου η αρχιτεκτονική φαίνεται στην παρακάτω εικόνα (βλ. Εικόνα 4-7).



Εικόνα 4-7: Αρχιτεκτονική LeNet (Lecun et al., 1998).

Όπως φαίνεται στην Εικόνα 4-7, η αρχιτεκτονική απαρτίζεται κυρίως από Convolutional Layers και Subsampling layers. Μετά την επιτυχία του LeNet στο να μπορεί να αναγνωρίσει ψηφία από εικόνες, ξεκίνησαν διάφοροι ερευνητές να δημιουργούν δικές τους αρχιτεκτονικές. Μερικές που αξίζει να

αναφερθούν είναι η AlexNet (Krizhevsky et al., 2012), VGG (Simonyan & Zisserman, 2015) και ResNet (He et al., 2015b).

#### 4.3.1. Χρήση Μηχανικής Όρασης στην Ιατρική

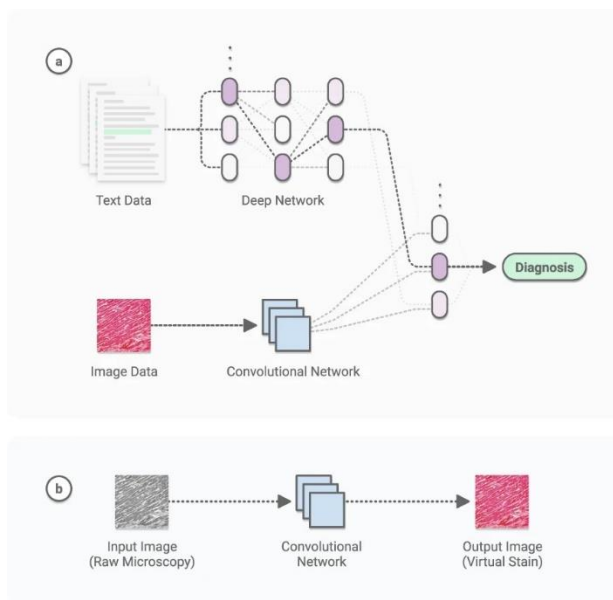
Τα Convolutional Networks χρησιμοποιούνται σε διάφορους κλάδους, μερικοί από αυτούς είναι self-driving cars (Barua et al., 2019), καθοδήγηση ρομπότ (Courtney et al., 1984), αναγνώριση προσώπου (Apoorva. et al., 2019) και στην Ιατρική (Gao et al., 2018). Επίσης η χρήση τους επεκτείνεται στην αρχαιολογία (Resler et al., 2021), γεωργία (Kamilaris & Prenafeta-Boldú, 2018), παρακολούθηση και ασφάλεια (Iqbal et al., 2021) και βιομηχανία (J. Wang et al., 2018).

Συγκεκριμένα στον κλάδο της Ιατρικής, έχουν δημιουργηθεί μοντέλα για διάφορες λειτουργίες, όπως ταξινόμηση κερατοειδούς μοτίβων (Haggag et al., 2019), αναγνώριση διαβητικής αμφιβληστροειδοπάθειας (Khalil et al., 2019), διάγνωση COVID-19 (Arias-Garzón et al., 2021, p. 19), Καρκίνο του μαστού (Zahoor et al., 2020) και δερματικών κακώσεων (Maglogiannis & Doukas, 2009).

Τα τελευταία χρόνια η Μηχανική Όραση έχει εφαρμοστεί σε πολλές έρευνες πάνω σε ιατρικές εικόνες (Litjens et al., 2017). Συγκεκριμένα στους τομείς της ραδιολογίας, παθολογίας, οφθαλμολογίας και δερματολογίας η μηχανική όραση χρησιμοποιείται ως εργαλείο αναγνώρισης προτύπων (pattern recognition) και ως διαγνωστικό εργαλείο.

Συγκεκριμένα, σε μία έρευνα (βλ. Εικόνα 4-8) έχει χρησιμοποιηθεί μηχανική όραση για τη διάγνωση καρκίνου του δέρματος όπου η απόδοση του μοντέλου ήταν ισότιμη με την απόδοση ειδικών ιατρών, δείχνοντας πως η τεχνητή νοημοσύνη έχει την ικανότητα και μπορεί να χρησιμοποιηθεί ως εργαλείο αναγνώρισης του καρκίνου του δέρματος (Esteva et al., 2017).

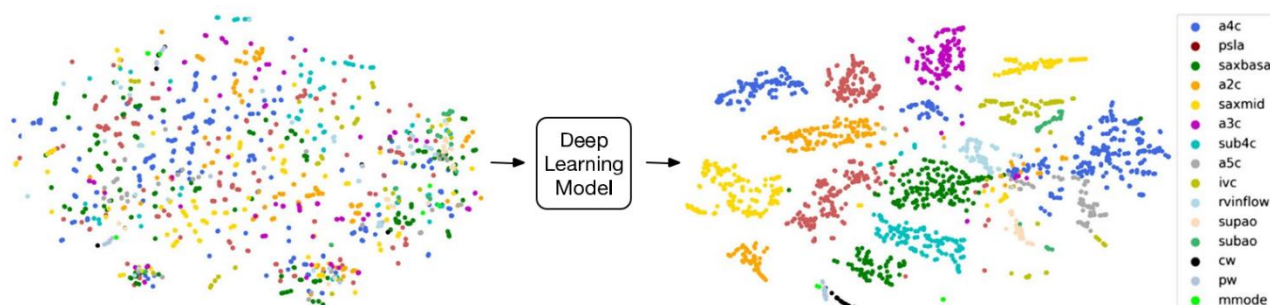
Παράλληλα ο σκοπός της έρευνας των (Gulshan et al., 2016) ήταν η αναγνώριση διαβητικής αμφιβληστροειδοπάθειας. Το μοντέλο βαθιάς μάθησης κατάφερε να διαγνώσει φωτογραφίες με ευαισθησία 97.5% και 96.1%. Επίσης αναφέρθηκε, πως η χρήση ενός αυτοματοποιημένου συστήματος



*Εικόνα 4-8: Δύο μοντέλα (a) και (b). Το πρώτο (a) παίρνει ως είσοδο εικόνες και κείμενο για να κάνει μία διάγνωση. Το (b) είναι ένα Generative Model για την αναπαραγωγή εικόνων. (Esteva et al., 2021)*

φέρνει συνέπεια στην ερμηνεία της εικόνας, υψηλά νούμερα ευαισθησίας και ακρίβειας. Τέλος το μοντέλο μπορεί σχεδόν στιγμιαία να δώσει τα αποτελέσματα της πρόβλεψης του, που αυτό μεταφράζεται σε πολύ γρήγορες διαγνώσεις (Gulshan et al., 2016).

Επιπλέον, επιστήμονες σχεδίασαν και εκπαιδύσαν ένα σύστημα βαθιάς μάθησης για την αναγνώριση 15 διαφορετικών ηχοκαρδιογραφημάτων (βλ. Εικόνα 4-9). 12 Εικόνες από “b-mode” και 3 από “Pulsed-wave doppler (PW)”, “continuous-wave Doppler (CW), και “m-mode”. Η ακρίβεια του μοντέλου ήταν κατά μέσο όρο 97.8% και ο χρόνος που χρειάστηκε για την ταξινόμηση (όχι εκπαίδευση) των εικόνων ήταν 21 ms ανά εικόνα σε ένα απλό υπολογιστή (Madani et al., 2018).



*Εικόνα 4-9: Ένα μοντέλο βαθιάς μάθησης ταξινομεί ταυτόχρονα 15 διαφορετικά ηχοκαρδιογραφήματα (Madani et al., 2018)*

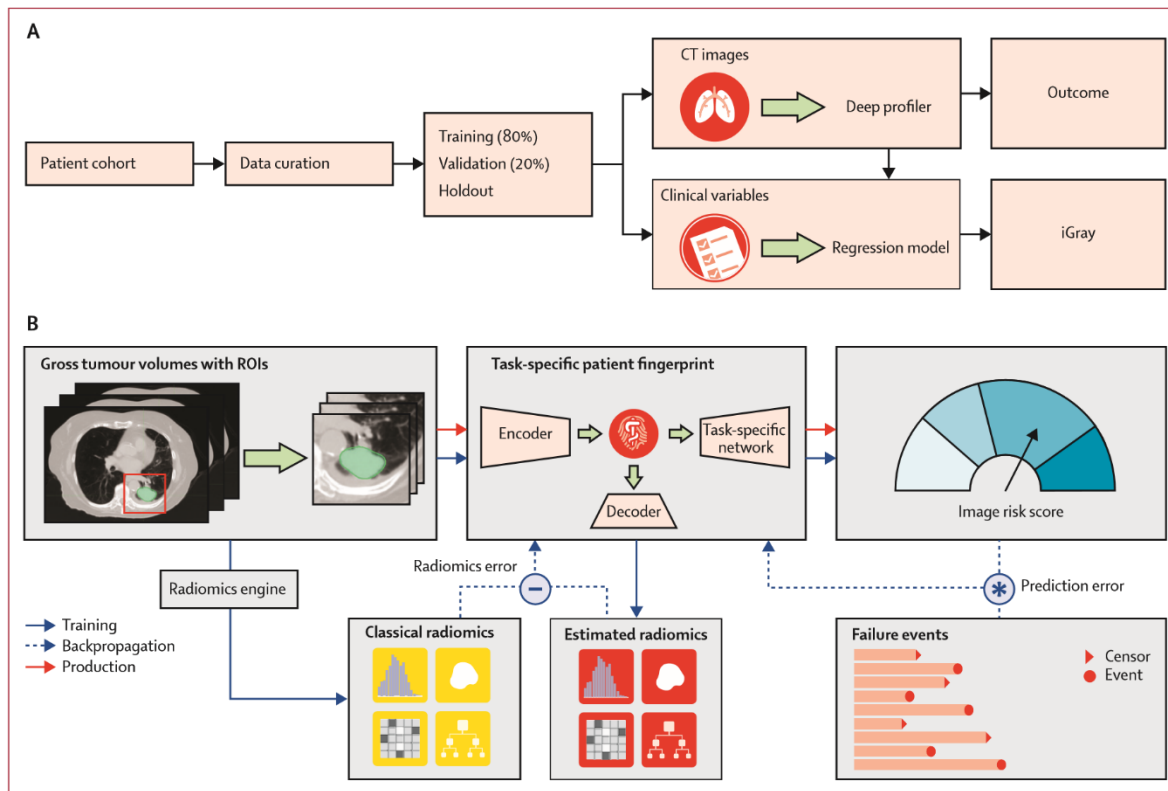
Ένας ακόμη επιστημονικός τομέας της Ιατρικής που μελετάτε το ανθρώπινο σώμα και τα συστήματά του είναι το Systems Medicine (SM), όπου σκοπός είναι να μας βοηθήσει να κατανοήσουμε, να αποτρέψουμε και να αντιμετωπίσουμε ιδιαίτερες και πολύπλοκες ασθένειες (H. Wang et al., 2021). Συγκεκριμένα, ο κλάδος αυτός ενσωματώνει σύνθετες βιοχημικές, ψυχολογικές και περιβαλλοντικές αλληλεπιδράσεις που διατηρούν ζωντανούς οργανισμούς με σκοπό την κατανόησή τους (Federoff & Gostin, 2009).

Σε ένα τέτοιο κλάδο, τα δεδομένα είναι πολυδιάστατα, ανομοιογενή και ο όγκος τους τεράστιος (Zanin et al., 2019). Για αυτό τον λόγο η χρήση βαθιάς μηχανικής μάθησης στον κλάδο Systems Medicine, δείχνει μεγάλη προοπτική (Chen et al., 2019).

Η εξατομικευμένη Ιατρική είναι ένα παρακλάδι του System Medicine. Σκοπός είναι, κατά τη διάγνωση, πρόγνωση ή πρόβλεψη της θεραπείας ενός ασθενή, να εξετασθούν συγκεκριμένοι μεμονωμένοι παράγοντες που σχετίζονται με τον συγκεκριμένο ασθενή. Δηλαδή, το επίκεντρο είναι ο ασθενής όπου εξατομικευμένες πληροφορίες σχετικά με το ιστορικό υγείας και τον τρόπο ζωής του ασθενή λαμβάνονται υπόψη για την πορεία της θεραπείας του (Flores et al., 2013; H. Wang et al., 2021). Αυτά τα δεδομένα είναι πολυδιάστατα, αδόμητα και ετερογενή (Kieseberg et al., 2014), αυξάνοντας την ανάγκη για χρήση βαθιάς μηχανικής μάθησης όπου μπορεί να αντιμετωπίσει προβλήματα μεγάλης πολυπλοκότητας (Flores et al., 2013).



Επιστήμονες ξεκίνησαν να εφαρμόζουν μεθόδους βαθιάς μηχανικής μάθησης για την λύση τέτοιων πολύπλοκων προβλημάτων. Για παράδειγμα, δημιουργήθηκε ένα CNN όπου βοηθάει με την ακτινοθεραπεία ασθενών που πάσχουν από εγκεφαλικό όγκο (Liu et al., 2019). Σε μία άλλη έρευνα, πρότειναν ένα CNN για πρόβλεψη κλινικής παρέμβασης μέσα σε μονάδες εντατικής θεραπείας (Suresh et al., 2017). Έπειτα, σε μία άλλη έρευνα (βλ. *Εικόνα 4-10*), κατάφεραν να δημιουργήσουν ένα σύστημα βαθιάς μάθησης σε συνδυασμό με κλινικούς παράγοντες κινδύνου ενός ασθενή, όπου προσαρμόζει την δόση ραδιενέργειας που θα λάβει το σώμα του, με αποτέλεσμα να μπορεί να παραδώσει εξατομικευμένες θεραπείες σε ασθενείς (Lou et al., 2019). Ακόμη μία έρευνα είναι η ανίχνευση υπό-ομάδων καρκίνου από γονιδιωματικά δεδομένα μέσω ενός multimodal deep belief network (DBN) (Liang et al., 2015).



*Εικόνα 4-10: Η δομή του συστήματος Deep Profiler (Lou et al., 2019).*

Μία ακόμη εφαρμογή της βαθιάς μηχανικής μάθησης είναι η διάγνωση πνευμονικών λοιμώξεων όπως πνευμονία, καρκίνο του πνεύμονα και φυματίωση με την χρήση ιατρικής απεικόνισης όπως αξονικές τομογραφίες ή ακτινογραφίες (Khan et al., 2021).

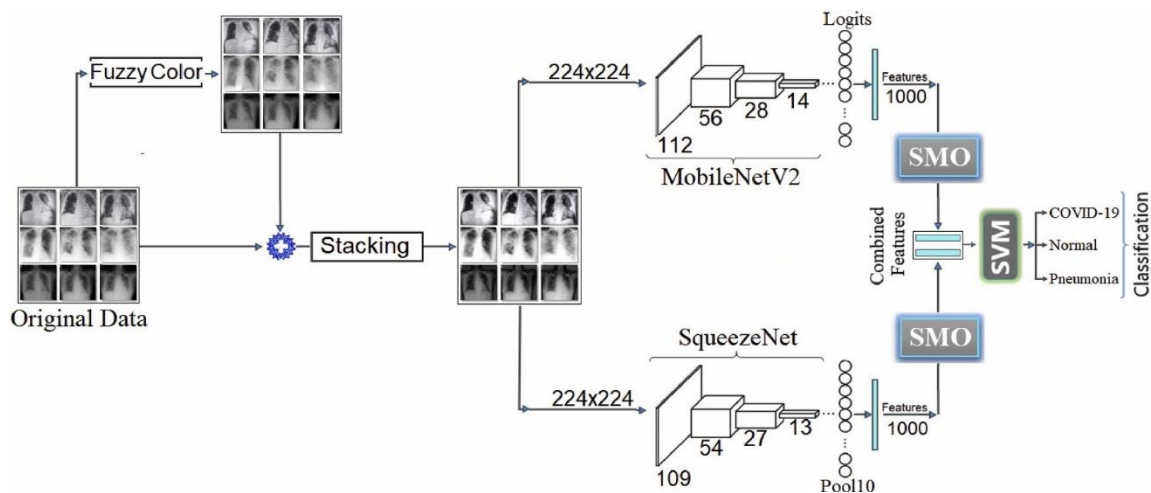
#### 4.3.2. Χρήση ακτινογραφιών σε συνδυασμό με μηχανική όραση

Όπως αναφέρθηκε στην *Εισαγωγή* της παρούσας πτυχιακής εργασίας, οι ακτινογραφίες σε σχέση με τις αξονικές τομογραφίες, έχουν χαμηλότερο κόστος, η αναπαραγωγή της εικόνας γίνεται γρηγορότερα, σαν μέθοδος είναι άμεσα διαθέσιμη στο κοινό και το σώμα εκτίθεται σε λιγότερη ραδιενέργεια (Brenner & Hall, 2007).

Οι ακτινογραφίες αποδείχθηκαν ως ένα πολύ σημαντικό εργαλείο για τη διαχείριση ασθενών. Συγκεκριμένα, με τη χρήση αξονικών ακτινογραφιών, μειώθηκαν τα παραπεμπτικά σε ειδικούς γιατρούς, και επίσης μειώθηκαν οι αλλαγές σε θεραπείες ασθενών. Αυτές οι αλλαγές στη διαχείριση ασθενών, φαίνονται πιο έντονα σε άτομα με βήχα, ή σε άτομα που παρουσίασαν σωματικές ανωμαλίες και τέλος σε άτομα με πιθανή διάγνωση πνευμονίας (Speets et al., 2006). Συμπληρωματικά, παρατηρήθηκε πως οι αξονικές φωτογραφίες προσφέρουν μεγάλη χρηστικότητα, και η πρόσβαση σε ένα τέτοιο διαγνωστικό εργαλείο συνεπάγεται σε εξοικονόμηση κόστους και χρόνου για γιατρούς και ασθενείς (Geitung et al., 1999).

### 5. Βιβλιογραφική ανασκόπηση

Στην έρευνα των (Toğazcar et al., 2020), δημιουργήθηκε ένα σύστημα για την ανίχνευση κορονοϊού, χρησιμοποιώντας τις αρχιτεκτονικές, MobileNetV2 και SqueezeNet. Η ταξινόμηση έγινε μέσω ενός Support Vector Machine (SVM). Το σετ δεδομένων της έρευνας ήταν χωρισμένο στις κατηγορίες: COVID-19, Normal και Pneumonia. Το μέγεθος των δεδομένων είναι στις 458 εικόνες συνολικά. Το 70% των εικόνων χρησιμοποιήθηκε για την εκπαίδευση και το 30% για την αξιολόγηση του συστήματος. Ως προεπεξεργασία των εικόνων χρησιμοποιήθηκε η τεχνική Fuzzy Color.



Εικόνα 5-1: Η ολοκληρωμένη δομή του συστήματος (Toğazcar et al., 2020).

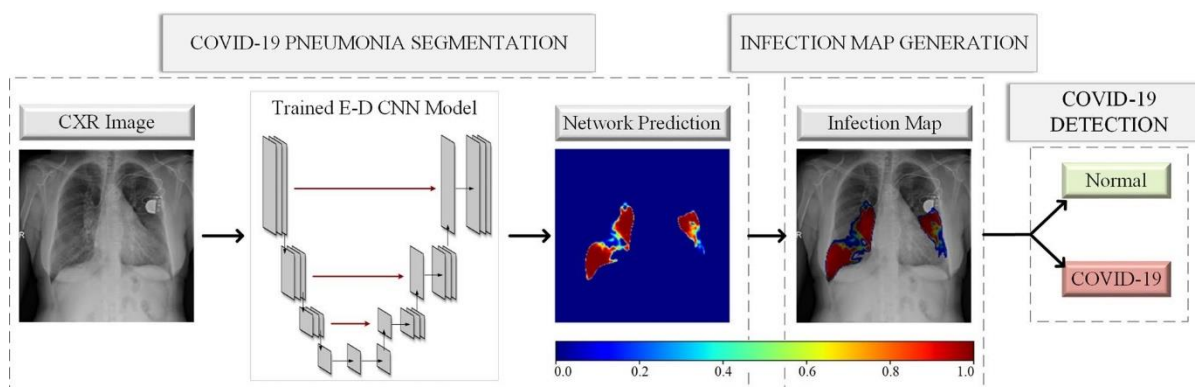


Αρχικά, το πρώτο βήμα είναι η επεξεργασία του σετ δεδομένων με τη μέθοδο Fuzzy Color, όπου αφαιρείτε ο θόρυβος της εικόνας. Έπειτα, στοιβάζονται οι αρχικές εικόνες με τις νέες Fuzzy Color εικόνες με σκοπό τη δημιουργία ενός νέου σετ δεδομένων με εικόνες καλύτερης ποιότητας. Στη συνέχεια τα δύο μοντέλα που αναφέρθηκαν, εκπαιδεύτηκαν πάνω στο νέο σετ δεδομένων και με τη χρήση του SMO αλγόριθμου, έγινε εξαγωγή χρήσιμων χαρακτηριστικών από τις εικόνες. Τέλος, για το κομμάτι της ταξινόμησης, τα χαρακτηριστικά καταχωρήθηκαν στο SVM.

Για την αξιολόγηση του συστήματος χρησιμοποιήθηκαν οι μετρήσεις *Sensitivity*, *Specificity*, *F-Score*, *Precision*, *Accuracy* και το *Confusion Matrix*. Τα αποτελέσματα του συστήματος είναι αρκετά υψηλά. Για εικόνες με κορονοϊό η ακρίβεια ήταν **99.34%**, για εικόνες χωρίς κορονοϊό ή πνευμονία, η ακρίβεια ήταν **98.68%** και τέλος για εικόνες με πνευμονία, η ακρίβεια ήταν **98.47%**. Σε γενικές γραμμές, η ακρίβεια του συστήματος ήταν **98.25%**.

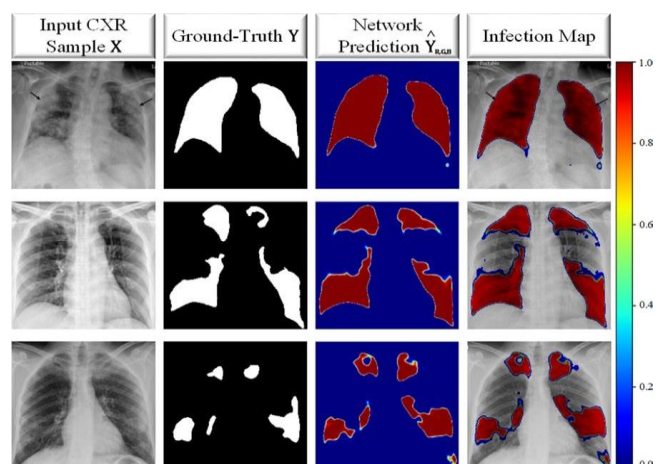
Παρόλα αυτά, το μέγεθος του σετ δεδομένων που έχει χρησιμοποιηθεί στην προκειμένη έρευνα, δεν είναι επαρκές. Συνολικά, στο σετ υπάρχουν 295 εικόνες για την κατηγορία COVID-19, 65 εικόνες για την κατηγορία Normal και 98 εικόνες για την κατηγορία Pneumonia. Πέρα του ότι το σετ δεν είναι ισορροπημένο, γενικότερα οι εικόνες σε κάθε κατηγορία δεν είναι αρκετές έτσι ώστε να μπορεί το μοντέλο να αναγνωρίζει σωστά εικόνες που δεν έχει ξαναδεί. Αυτό μπορεί να παρουσιάσει πρόβλημα επειδή το σύστημα μαθαίνει πολύ καλά το υφιστάμενο σετ δεδομένων και για αυτό έχει υψηλή ακρίβεια. Αλλά δεν σημαίνει πως θα έχει την ίδια απόδοση σε δεδομένα που δεν έχει ξαναδεί. Επίσης, για τη δημιουργία του σετ δεδομένων, πάρθηκαν εικόνες από 3 διαφορετικές πηγές. Οι (Maguolo & Nanni, 2020) στην ερευνά τους έδειξαν πως ένα σύστημα μπορεί να μάθει χαρακτηριστικά σχετικά με την πηγή των εικόνων και όχι σχετικά με την ιατρική πληροφορία που παρέχουν οι εικόνες. Στην προκειμένη περίπτωση, εικόνες που σχετίζονται με COVID-19 πάρθηκαν από 2 διαφορετικές πηγές και για τις κατηγορίες Normal και Pneumonia οι εικόνες πάρθηκαν από την τρίτη πηγή. Αυτό μπορεί να παρουσιάσει πρόβλημα εφόσον το μοντέλο έχει τη δυνατότητα να αναγνωρίζει από ποια πηγή σχετίζεται η ανάλογη κλάση, και η διάγνωση που κάνει, μπορεί να βασίζεται στην πηγή της εικόνας παρά στα ιατρικά χαρακτηριστικά της. Τέλος αξίζει να σχολιαστεί πως ενδεικτικά, η απόδοση του συστήματος είναι καλή, και εξαιτίας της αρχιτεκτονικής MobileNetV2, το σύστημα μπορεί να τρέξει σε κινητές συσκευές. Δυστυχώς, όταν έγινε η συγκεκριμένη έρευνα (Toğaçar et al., 2020), δεν υπήρχαν αρκετές αξονικές ακτινογραφίες για κορονοϊό και ίσως για αυτό τον λόγο, οι εικόνες που χρησιμοποιήσαν να ήταν ανεπαρκείς.

Οι (Degerli et al., 2021) στην ερευνά τους, δημιούργησαν ένα μοντέλο βαθιάς μάθησης το οποίο μπορεί να εντοπίσει σε πιο σημείο του πνεύμονα δημιουργήθηκε μόλυνση. Επίσης έχει την ικανότητα να βαθμολογήσει την σοβαρότητα της μόλυνσης και να αναγνωρίσει αν προέρχεται από κορονοϊό, παράγοντας ένα *infection map* για κάθε εικόνα. Για την παραγωγή του *infection map*, άνθρωποι με τη βοήθεια υπολογιστών δημιούργησαν *ground-truth-segmentation masks* για κάθε εικόνα, έτσι ώστε να μπορέσει το μοντέλο να μάθει πως να ξεχωρίζει τους πνεύμονες και τα σημεία που παρατηρείται η λοίμωξη.



Εικόνα 5-2: Το σύστημα που δημιούργησαν οι (Degerli et al., 2021) για την αναγνώριση κορονοϊού.

Το σετ δεδομένων που χρησιμοποιήθηκε στην προκειμένη έρευνα προέρχεται από διάφορες πηγές. Για αυτό τον λόγο, υπήρχαν μερικές όμοιες εικόνες ή εικόνες χαμηλής ποιότητας οι οποίες αναγνωρίστηκαν και αφαιρέθηκαν. Συνολικά, ο αριθμός των εικόνων ανέρχεται σχεδόν στις 120,000 και 2951 από αυτές σχετίζονται με κορονοϊό. Οι εικόνες είναι ταξινομημένες σε 2 κατηγορίες, Normal και COVID-19. Όσο αφορά την αξιολόγηση του συστήματος, χρησιμοποιήθηκαν διαφορετικές εικόνες οι οποίες είναι χωρισμένες σε 2 ομάδες, “Control Group 1” και “Control Group 2”. Το “Control Group 1” απαρτίζεται από εικόνες χωρίς κάποια λοίμωξη, δηλαδή οι πνεύμονες είναι υγιείς. Αντιθέτως, το “Control Group 2” απαρτίζεται από εικόνες 15 διαφορετικών λοιμώξεων. Μερικές λοιμώξεις του “Group 2” είναι



Εικόνα 5-3: Παράδειγμα δημιουργίας του *infection map* (Degerli et al., 2021).

ατελεκτασία, καρδιομεγαλία, πνευμονία, πνευμοθώρακα, εμφύσημα και υγιής.

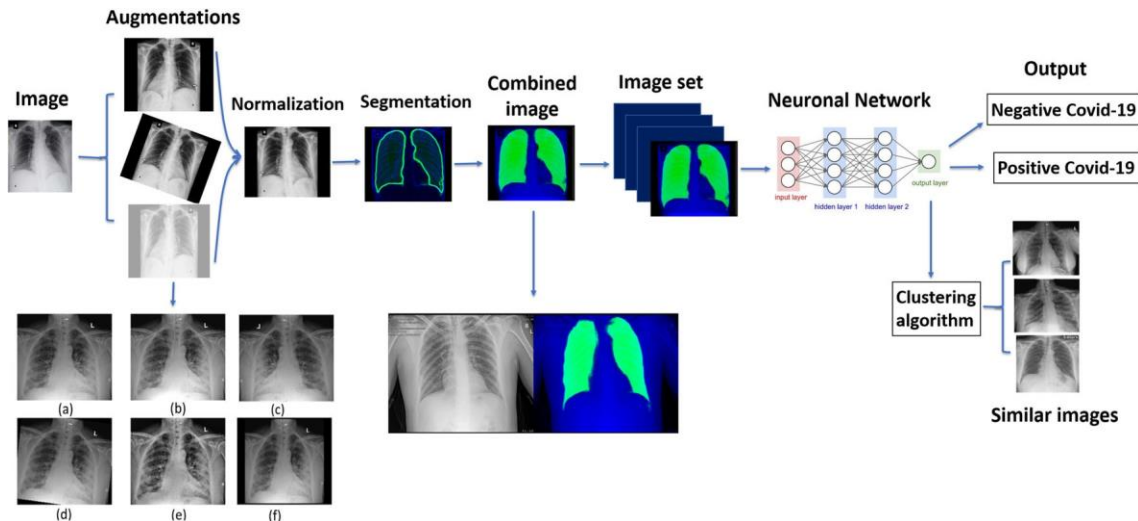
Όσο αφορά το segmentation των εικόνων, χρησιμοποιήθηκαν μοντέλα βασισμένα στην αρχιτεκτονική *U-Net* (Ronneberger et al., 2015), *UNet++* (Zhou et al., 2018) και *DLA* (Yu et al., 2018). Το επόμενο βήμα είναι να εφαρμοστεί segmentation στη περιοχή που ανιχνεύεται η λοίμωξη δημιουργώντας ένα *infection map*, ακολούθως χρησιμοποιείται το *infection map* για εντοπισμό κορονοϊού χρησιμοποιώντας μοντέλα

βασισμένα στις αρχιτεκτονικές *DenseNet-121* (G. Huang et al., 2018), *CheXNet* (Rajpurkar et al., 2017), *Inception-V3* (Szegedy et al., 2015) και *ResNet-50* (He et al., 2015b).

Τα αποτελέσματα της συγκεκριμένης έρευνας (Degerli et al., 2021) δείχνουν να είναι αξιόπιστα έχοντας απόδοση **81.72% sensitivity** και **83.20% F1-Score** στον εντοπισμό της λοίμωξης. Όσο αφορά τον εντοπισμό κορονοϊού, η απόδοση του συστήματος ήταν **94.96% sensitivity** και **99.88% specificity**.

Συμπερασματικά, η μεθοδολογία και τα αποτελέσματα που παρουσίασαν φαίνονται αξιόπιστα. Επίσης το μέγεθος του σετ δεδομένων ήταν επαρκής και η απόδοση του συστήματος ήταν τεκμηριωμένη μέσα από εκτενή έλεγχο. Παράλληλα, το σύστημα μπορεί να αναγνωρίσει τον κορονοϊό ανάμεσα από διάφορες ασθένειες του θώρακα, δημιουργώντας infection maps σε κάθε εικόνα, κάτι που μπορεί να συνδράμει στην αντικειμενική εκτίμηση των γιατρών στο κορονοϊό.

Στην έρευνα των (Keidar et al., 2021) χρησιμοποιήθηκαν 5 μοντέλα (*ResNet34*, *ResNet50*, *ResNet152*, *VGG16*, και *CheXpert*) για την αναγνώριση κορονοϊού ανάμεσα από 2426 αξονικές ακτινογραφίες. Παράλληλα, εφαρμόστηκε “lung segmentation” στις εικόνες μέσω του μοντέλου *U-Net*. Επιπρόσθετα, δοκίμασαν να συνδυάσουν και τα 5 μοντέλα για αναγνώριση κορονοϊού (ensemble method). Πέρα από την αναγνώριση κορονοϊού, το σύστημα μπορεί να ομαδοποιήσει παρόμοιες εικόνες σε σχέση με την εικόνα εισόδου.



Εικόνα 5-3: Το ολοκληρωμένο σύστημα της προκυμμένης έρευνας (Keidar et al., 2021).

Όσο αφορά την μεθοδολογία, οι (Keidar et al., 2021) χρησιμοποίησαν τα μοντέλα που αναφέρθηκαν παραπάνω σε συνδυασμό με *Transfer Learning* (Bozinovski, 2020) και *lung segmentation*. Επιπρόσθετα, χρησιμοποίησαν ένα μοντέλο συνόλου (ensemble model), όπου έχει ως έξοδο τον μέσο όρο των αποτελεσμάτων από τα 5 μοντέλα.

Ακολούθως, με σκοπό τη μεγιστοποίηση του σετ δεδομένων, χρησιμοποιήθηκε η μέθοδος *Data Augmentation*. Στην εν λόγω μέθοδο, γίνονται μικρές μετατροπές σε κάθε εικόνα εμπλουτίζοντας το σετ δεδομένων. Αυτό συνήθως αυξάνει την απόδοση του μοντέλου εφόσον υπάρχει περισσότερο υλικό για εκμάθηση. Στην προκειμένη έρευνα, οι μετατροπές που εφαρμόστηκαν σε κάθε εικόνα είναι, α) αλλαγή στην φωτεινότητα, β) οριζόντια αναστροφή, γ) περιστροφή κατά 7°, δ) CLAHE και τέλος ε) αλλαγή στην κλίμακα. Στην συνέχεια, εκπαιδεύτηκαν και αξιολογήθηκαν τα 5 μοντέλα που αναφέρθηκαν παραπάνω.

Για την αξιολόγηση του συστήματος, χρησιμοποιήθηκαν οι μετρήσεις, *accuracy*, *sensitivity*, *specificity*, *AUC*, *ROC*, *precision / Recall* και *Confidence Intervals*. Από τα 5 μοντέλα που χρησιμοποιήθηκαν, το *ResNet50* είχε την καλύτερη απόδοση με *accuracy* 90.0%, *Sensitivity* 90.5% και *Specificity* 89.4%. Συμπληρωματικά, το μοντέλο συνόλου είχε την καλύτερη απόδοση με *accuracy* 90.3%, *Sensitivity* 90.5% και *Specificity* 90.0%.

Αξίζει να αναφερθεί πως οι εικόνες που χρησιμοποιήθηκαν προέρχονται από το ίδιο μηχάνημα. Αυτό είναι αρκετά σημαντικό διότι σύμφωνα με τους (Maguolo & Nanni, 2020) ένα μοντέλο βαθιάς μάθησης μπορεί να μάθει να αναγνωρίζει τις εικόνες ως προς την πηγή τους, παρά από τα ιατρικά χαρακτηριστικά τους. Επιπρόσθετα, η ικανότητα του συστήματος στο να βρίσκει παρόμοιες εικόνες μέσα στο σετ δεδομένων είναι ένα πλεονέκτημα διότι προσφέρει στους γιατρούς ένα σημείο αναφοράς για την κατάσταση της ασθένειας σε σχέση με παρόμοια περιστατικά.

Οι (Narin et al., 2021) στην έρευνα τους εκπαιδυσαν 5 μοντέλα σε 3 διαφορετικά σετ δεδομένων. Συγκεκριμένα τα μοντέλα που χρησιμοποιήθηκαν για την αναγνώριση κορονοϊού είναι τα *ResNet50*, *ResNet101*, *ResNet152*, *InceptionV3* και *Inception-ResNetV2*.

Όσο αφορά τα δεδομένα που χρησιμοποιήθηκαν (βλ. πίνακα 5-1), και τα 3 σετ περιλαμβάνουν 341 εικόνες με την κλάση “COVID-19”. Ακολούθως, το 1°, 2° και 3° σετ, αποτελείται από 2800 εικόνες για την κατηγορία *Normal*, 1493 εικόνες για την κατηγορία *Viral Pneumonia* και τέλος, 2772 εικόνες για την κατηγορία *Bacterial Pneumonia* αντίστοιχα.

*Πίνακας 5-1: Τα περιεχόμενα του κάθε σετ δεδομένων (Narin et al., 2021)*

Datasets	Bacterial pneumonia	COVID-19	Normal	Viral pneumonia
Dataset-1	—	341	2800	—
Dataset-2	—	341	—	1493
Dataset-3	2772	341	—	—

Αναφορικά με την μεθοδολογία των (Narin et al., 2021), χρησιμοποιήθηκε η μέθοδος *Transfer Learning*. Τα 5 μοντέλα αξιολογήθηκαν και στα 3 σετ δεδομένων, όπου το κάθε σετ χωρίστηκε τυχαία σε *train* και *test* με ποσοστά 80% και 20% αντίστοιχα. Παράλληλα χρησιμοποιήθηκε η μέθοδος *Cross-Validation*. Τέλος, οι μετρήσεις αξιολόγησης που πάρθηκαν για κάθε μοντέλο είναι, *Accuracy*, *Recall*, *Specificity*, *Precision*, και *F1-Score*. Ακολούθως, από τα 5 μοντέλα, το *ResNet50* είχε την καλύτερη απόδοση, όπου η ακρίβεια του στο 1<sup>ο</sup> σετ δεδομένων ήταν 96.1%, στο 2<sup>ο</sup> 99.5% και στο 3<sup>ο</sup> 99.7%.

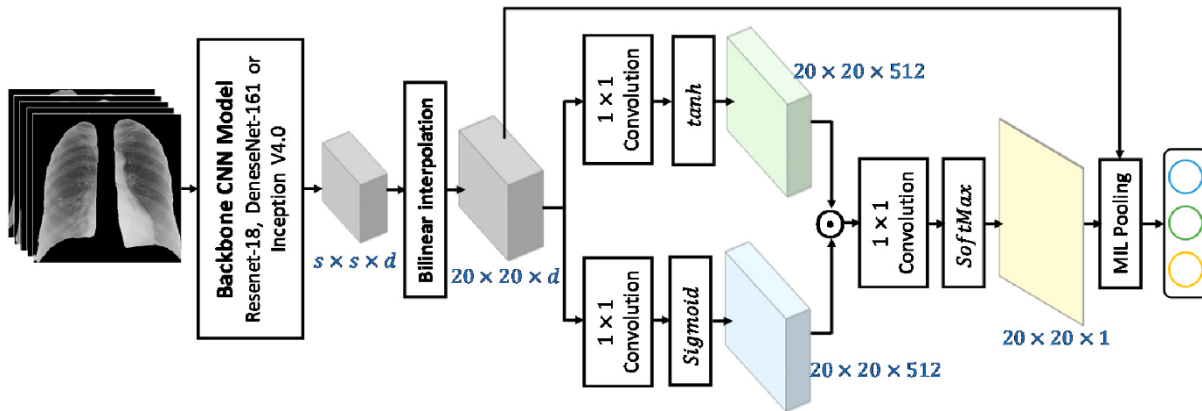
Βέβαια, πρέπει να σημειωθεί η ασυμμετρία και το μέγεθος των δεδομένων της έρευνας. Συγκεκριμένα, και στις 3 περιπτώσεις (σετ 1<sup>ο</sup>, 2<sup>ο</sup>, και 3<sup>ο</sup>), οι εικόνες που ανήκουν στην κατηγορία *COVID-19* είναι λιγότερες σε σχέση με τις άλλες κατηγορίες. Για παράδειγμα στο 1<sup>ο</sup> σετ (βλ. πίνακα 5-1), από τις συνολικά 3,141 εικόνες, το ποσοστό που αναλογεί στην κατηγορία *COVID-19* είναι μόνο 10.85%, το υπόλοιπο 89.14% αντιστοιχεί στην κατηγορία *Normal*. Αυτό προκαλεί ανισορροπία στα αποτελέσματα του μοντέλου, κάτι που φαίνεται από τα αποτελέσματα του 1<sup>ου</sup> σετ, όπου το *ResNet50* έχει *Recall* 91.8% αλλά *Precision* 76.5%. Βέβαια αξίζει να σημειωθεί πως το μοντέλο, έφερε καλύτερα αποτελέσματα στο 2<sup>ο</sup> και 3<sup>ο</sup> σετ δεδομένων, παρόλο που παραμένει το αρχικό πρόβλημα της ασυμμετρίας. Δυστυχώς, στην προκειμένη έρευνα, δεν έγινε αξιολόγηση του συστήματος σε δεδομένα που δεν έχει ξαναδεί. Καλή πρακτική θα ήταν, να υπήρχε *train set*, *validation set* και τέλος, *test set*, όπου το σύστημα εκπαιδεύεται πάνω στο *train set*, αξιολογείτε η απόδοση της εκμάθησης του πάνω στο *validation set*, και αφού τα αποτελέσματα του *validation set* είναι αποδεκτά, τότε γίνεται μία τελική αξιολόγηση στο *test set*. Η συγκεκριμένη πρακτική θα πρόσφερε στους ερευνητές μία καλύτερη επίγνωση, για την αποδοτικότητα του συστήματος τους. Η ονομασία του κάθε σετ δεν παίζει μεγάλο ρόλο, το σημαντικό είναι πως το σύστημα δεν αξιολογήθηκε σε δεδομένα που δεν έχει ξαναδεί. Αδιαμφισβήτητα, είναι εξαιρετικά σημαντική η διεξοδική ανάλυση των αποτελεσμάτων ενός συστήματος, όπου σκοπός του είναι η λήψη αποφάσεων σχετικά με την υγεία ενός ασθενή.

Στην έρευνα των (Afifi et al., 2021), παρουσιάστηκε ένα σύνολο μοντέλων βαθιάς μάθησης, για την ανίχνευση κορονοϊού. Συνολικά, το σετ δεδομένων περιέχει 11,197 αξονικές ακτινογραφίες, όπου 1056 αντιστοιχούν στην κατηγορία *COVID-19*, 5451 αντιστοιχούν στην κατηγορία *Pneumonia* και τέλος 7217 αντιστοιχούν στην κατηγορία *Control*. Η κατηγορία *Control* απαρτίζεται από ακτινογραφίες με άλλες πνευμονολογικές ασθένειες (εκτός *COVID-19* και *Pneumonia*) ή ακτινογραφίες με καμία πάθηση (υγιής). Στη συνέχεια, εφαρμόστηκε *lung localization* και *lung segmentation* στο σετ δεδομένων. Έπειτα, το σετ δεδομένων χωρίστηκε σε *training*, *validation* και *testing* με ποσοστά 70%, 15% και 15% αντίστοιχα (βλ. πίνακα 5-2). Τα 5 καλύτερα μοντέλα στην απόδοση εκμάθησης, αξιολογούνται στην τελική φάση πάνω στο *test set*.

Πίνακας 5-2: Ο κατακερματισμός του σετ δεδομένων σε training, validation και test (Afifi et al., 2021).

Class	Training	Validation	Test
Control	5009	1076	1133
Pneumonia	3934	822	785
COVID19	739	158	159

Όσο αφορά τη μεθοδολογία της συγκεκριμένης έρευνας, το πρώτο βήμα ήταν να εφαρμοστεί *lung localization* με τη χρήση του μοντέλου *Retina-Net* (T.-Y. Lin et al., 2017) και *lung segmentation* με την χρήση ενός διαμορφωμένου μοντέλου *UNet* και *ENet*. Συγκεκριμένα, η μέθοδος *lung segmentation* διευκολύνει το μοντέλο στο να μαθαίνει συναφής χαρακτηριστικά με τον κορονοϊό. Σύμφωνα με τους (Yasin & Gouda, 2020), ο κορονοϊός και η πνευμονία έχουν τα ίδια βασικά συμπτώματα, τα οποία μπορούν να εμφανιστούν σε ολόκληρους του πνεύμονες, ή μόνο σε τοπικά σημεία. Για αυτό τον λόγο, οι (Afifi et al., 2021) δημιούργησαν ένα σύστημα όπου μπορεί να χρησιμοποιήσει *global* και *local* χαρακτηριστικά.



Εικόνα 5-4: Η δομή του συστήματος (Afifi et al., 2021).

Για το κομμάτι της αναγνώρισης κορονοϊού, αξιοποίησαν 3 μοντέλα με διαφορετικές αρχιτεκτονικές. Συγκεκριμένα, χρησιμοποίησαν *ResNet18*, *Inception v4.0*, και *DenseNet161* μαζί με *multi-instance polling* το οποίο μπορεί να εξάγει τοπικά χαρακτηριστικά από τις εικόνες. Έπειτα, κατά την διάρκεια της εκπαίδευσης, εφαρμόστηκαν μερικές τροποποιήσεις στις εικόνες (*Data Augmentation*), όπως οριζόντια αναστροφή, περιστροφή από  $-15^\circ$  ως  $15^\circ$  και τροποποίηση φωτεινότητας (κλίμακας 0.9 με 1.1). Τέλος, για την αποφυγή του *Overfitting* εφαρμόστηκε η μέθοδος *early stopping* όπου σταματάει η εκπαίδευση εάν δεν υπάρξει σημαντική βελτίωση στο *validation loss* για 5 μεταγενέστερα *epochs*.

Σχετικά με την απόδοση του συστήματος, χρησιμοποιήθηκαν οι μετρήσεις *AUC*, *ROC*, *Geometric mean*, *balanced accuracy*, *average precision score*, και τέλος *F1-Score*. Σε ότι αφορά το κομμάτι του *lung segmentation*, χρησιμοποιήθηκε η μέτρηση *Dice Score Coefficient*, όπου το μοντέλο υπεύθυνο για αυτό το

κομμάτι πέτυχε **95%  $\pm$  2.7**. Τα αποτελέσματα του κάθε μοντέλου σχετικά την αναγνώριση κορονοϊού φαίνονται στον παρακάτω πίνακα (βλ. Πίνακα 5-3).

Το *DenseNet161*, βάση των αποτελεσμάτων φαίνεται να έχει την καλύτερη απόδοση συγκριτικά με το *ResNet18* και *Inception v4.0*. Συγκεκριμένα, το *DenseNet161* στην αναγνώριση κορονοϊού και πνευμονίας είχε κατά μέσο όρο ακρίβεια **97.5%** και **92.1%** αντίστοιχα.

Στη συνέχεια, ακολούθησε στατιστική ανάλυση των μοντέλων, με σκοπό την εύρεση σημαντικών διαφορών μεταξύ τους. Για την ακρίβεια, χρησιμοποιήθηκε το Friedman test (Demšar, 2006) και το post-Hoc Nemenyi test (Nemenyi, 1963) για τη σύγκριση των μοντέλων. Παράλληλα, χρησιμοποιήθηκε η μέθοδος *occlusion technique* (Zeiler & Fergus, 2014).

**Πίνακας 5-3: Η απόδοση του κάθε μοντέλου χρησιμοποιώντας *Micro-average* (Afifi et al., 2021).**

CNN Model	Evaluation Metric	Global Only	Local Attention	Ensemble
ReseNet18	Balanced Accuracy	86.8% (85.0, 88.9)	87.2% (84.9, 89.5)	88.7% (88.8, 91.3)
	Average Precession	87.9% (86.2, 89.9)	87.3% (85.0, 89.6)	88.8% (86.4, 91.4)
	F1 Score	87.3% (85.6, 89.3)	88.5% (86.2, 90.8)	89.3% (87.1, 91.7)
DenseNet161	Balanced Accuracy	88.9% (87.2, 90.9)	89.4% (87.2, 91.6)	91.2% (89.2, 93.4)
	Average Precession	90.7% (89.3, 92.4)	91.3% (89.3, 93.3)	92.4% (90.5, 94.6)
	F1 Score	89.6% (88.1, 91.4)	90.2% (88.0, 92.4)	91.9% (89.9, 94.1)
Inception V4.0	Balanced Accuracy	84.7% (82.9, 86.8)	86.2% (83.7, 88.7)	87.5% (85.1, 90.1)
	Average Precession	85.5% (83.7, 87.6)	88.7% (86.4, 91.0)	88.9% (86.5, 91.5)
	F1 Score	86.3% (84.5, 88.4)	87.3% (85.0, 89.6)	88.2% (85.8, 90.8)

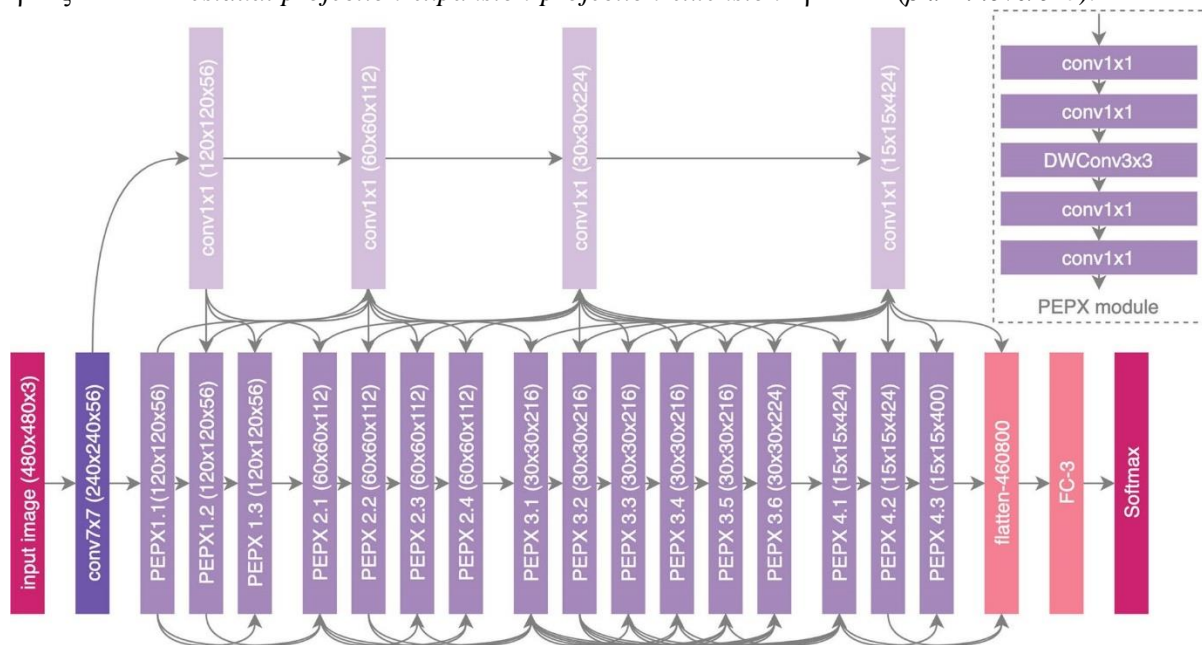
Ανακεφαλαιώνοντας, οι (Afifi et al., 2021) παρουσίασαν ένα σύστημα το οποίο μπορεί να διακρίνει α) ασθένειες όπως πνευμονία (είτε από κορονοϊό ή βακτηριακή), β) υγιές περιπτώσεις και γ) άλλες πνευμονολογικές ασθένειες. Στη συνέχεια, η χρήση του *lung segmentation* διευκολύνει στο πρόβλημα όπου ένα μοντέλο μπορεί να μάθει χαρακτηριστικά τα οποία δεν είναι σχετικά με την πάθηση, και να βασίζεται τα αποτελέσματα του σε αυτά τα χαρακτηριστικά. Το εν λόγω πρόβλημα συζητήθηκε στην έρευνα των (DeGrave et al., 2021). Έπειτα η χρήση *local* και *global* χαρακτηριστικών έδειξε αποτελεσματική, αυξάνοντας την ικανότητα του μοντέλου στο να διακρίνει τις περιπτώσεις με κορονοϊό. Τέλος, η χρήση του *Data Augmentation* δείχνει να βοήθησε στο πρόβλημα της δυσαναλογίας των κατηγοριών, αφού η κατηγορία *COVID-19* περιείχε μόνο 739 εικόνες για εκπαίδευση (βλ. Πίνακα 5-2). Η προκειμένη έρευνα



παρουσίασε μία αξιόλογη μέθοδο για τη καταπολέμηση του κορονοϊού, χρησιμοποιώντας σωστές πρακτικές στον χώρο της μηχανικής μάθησης. Κατά συνέπεια, τα αποτελέσματα τους είναι αξιόπιστα, εφόσον έγινε στατιστική ανάλυση του κάθε μοντέλου όσο αφορά την απόδοσή τους.

Στην έρευνα των (L. Wang et al., 2020) δημιουργήθηκε μία καινούρια CNN αρχιτεκτονική με το όνομα *COVID-Net*. Επιπρόσθετα, δημιούργησαν ένα δημόσιο σετ δεδομένων (*COVIDx*), όπου θα γίνει η εκπαίδευση του *COVID-Net*. Συγκεκριμένα το σετ δεδομένων απαρτίζεται από 13,975 ακτινογραφίες. Για την δημιουργία του σετ, συνδυάστηκαν και τροποποιήθηκαν 5 δημόσιες αποθήκες δεδομένων. Το συγκεκριμένο σετ είναι ελεύθερης πρόσβασης. Αυτό σημαίνει πως ο κάθε ερευνητής μπορεί να το χρησιμοποιήσει για την ερευνά του. Πέρα από αυτό, παρατηρείται πως στα περισσότερα σετ δεδομένων, υπάρχει περιορισμένος αριθμός εικόνων που σχετίζονται με κορονοϊό. Αυτό ισχύει και για το συγκεκριμένο σετ, καθώς υπάρχουν μόνο 358 εικόνες κορονοϊού. Όσο αφορά τις υπόλοιπες κατηγορίες, το σετ απαρτίζεται από 8,066 εικόνες χωρίς πνευμονία (normal) και 5,538 εικόνες με πνευμονία (χωρίς κορονοϊό).

Σχετικά με την αρχιτεκτονική *COVID-Net*, μπορεί να παρατηρηθεί πως ο σχεδιασμός της, απαρτίζεται κυρίως από ένα *residual projection-expansion-projection-extension* πρότυπο (βλ. Εικόνα 5-6).



Εικόνα 5-5: Η αρχιτεκτονική *COVID-Net* (L. Wang et al., 2020).

Αρχικά, η αρχιτεκτονική εκπαιδεύτηκε πάνω στο σετ ImageNet (Krizhevsky et al., 2012) και στην συνέχεια πάνω στο σετ *COVIDx*. Στη συνέχεια, ακολούθησε η ποιοτική ανάλυση *GSInquire* (Z. Q. Lin et al., 2019), με σκοπό να μάθουν αν η αρχιτεκτονική βασίζει τα αποτελέσματα που παράγει, σε σχετικά ιατρικά χαρακτηριστικά ή σε ασήμαντα που δεν έχουν να κάνουν με την ιατρική απεικόνιση της ακτινογραφίας.



Όσο αφορά την προ - επεξεργασία του σετ, όλες οι εικόνες κόπηκαν κατά 8% στο πάνω μέρος με σκοπό τη διαγραφή ενσωματωμένης κειμενικής πληροφορίας που βρίσκεται συνήθως στις ακτινογραφίες. Στη συνέχεια, χρησιμοποιήθηκε η μέθοδος *Data Augmentation* όπου έγιναν οι μερικές τροποποιήσεις σε κάθε εικόνα. Συγκεκριμένα, εφαρμόστηκε α) image translation ( $\pm 10\%$  σε x και y άξονα), β) περιστροφή ( $\pm 10^\circ$ ), γ) οριζόντια αναστροφή ( $\pm 15^\circ$ ), και δ) αλλαγή έντασης ( $\pm 10^\circ$ ).

Ακολούθως, έγινε ποσοτική ανάλυση της αρχιτεκτονικής. Συγκεκριμένα, υπολογίστηκε το *test accuracy*, *sensitivity*, και *positive predictive value (PPV)* για κάθε κατηγορία του σετ δεδομένων. Θα ακολουθήσει σχολιασμός των αποτελεσμάτων της έρευνας (L. Wang et al., 2020) ως προς την απόδοση της αρχιτεκτονικής COVID-Net.

**Πίνακας 5-4: Απόδοση του COVID-Net συγκριτικά με τις αρχιτεκτονικές VGG-19 και ResNet-50 (L. Wang et al., 2020).**

Architecture	Params (M)	MACs (G)	Acc. (%)
VGG-19	20.37	89.63	83.0
ResNet-50	24.97	17.75	90.6
COVID-Net	11.75	7.50	93.3

**Πίνακας 5-5: Sensitivity του COVID-Net συγκριτικά με τις αρχιτεκτονικές VGG-19 και ResNet-50 (L. Wang et al., 2020).**

Architecture	Normal	Non-COVID19	COVID-19
Sensitivity (%)			
VGG-19	98.0	90.0	58.7
ResNet-50	97.0	92.0	83.0
COVID-Net	95.0	94.0	91.0

**Πίνακας 5-6: PPV του COVID-Net συγκριτικά με τις αρχιτεκτονικές VGG-19 και ResNet-50 (L. Wang et al., 2020).**

Architecture	Normal	Non-COVID19	COVID-19
Positive predictive value (%)			
VGG-19	83.1	75.0	98.4
ResNet-50	88.2	86.8	98.8
COVID-Net	90.5	91.3	98.9

Παρατηρείται, πως το COVID-Net επιτυγχάνει 93.3% *test accuracy*, 91% *sensitivity* και 98.9% *PPV* για την κατηγορία *COVID-19* (βλ. Πίνακα 5-4, 5-5, 5-6). Συγκεκριμένα, ο υψηλός αριθμός *PPV* για την κατηγορία *COVID-19* είναι εξαιρετικά σημαντικός διότι υποδηλώνει πως η αρχιτεκτονική δεν πάσχει από

μεγάλο αριθμό *ψευδή θετικών*. Επίσης, όσο αφορά το 91% *Sensitivity* του μοντέλου για την κατηγορία *COVID-19*, στην προκειμένη περίπτωση υποδηλώνει πως δεν του διαφεύγουν σε μεγάλο βαθμό, εικόνες με κορονοϊό.

Όσο αφορά την πολυπλοκότητα της αρχιτεκτονικής, παρατηρείται πως το *COVID-Net* περιέχει λιγότερες παραμέτρους από τις αρχιτεκτονικές *ResNet50* και *VGG-19*. Συγκεκριμένα, το *COVID-Net* απαρτίζεται από 11,75 εκατομμύρια παραμέτρους ενώ το *ResNet50* και *VGG-19* από 24,97 εκατομμύρια και 20.37 εκατομμύρια παραμέτρους αντίστοιχα. Παρόλα αυτά, η αποδοτικότητα του *COVID-19* ξεπέρασε τις εν λόγω αρχιτεκτονικές στην αναγνώριση κορονοϊού.

Η ποιοτική ανάλυση *GSInquire* αποδεικνύει πως η αρχιτεκτονική βασίζει τα αποτελέσματα της σε χαρακτηριστικά που συνάδουν με την ιατρική απεικόνιση της ακτινογραφίας. Συγκεκριμένα φαίνεται πως εστιάζει σε οπτικούς παράγοντες που χρησιμοποιούν ιατροί για τη διάγνωση κορονοϊού όπως, *ground-glass opacities*, *bilateral abnormalities* και *interstitial abnormalities*.

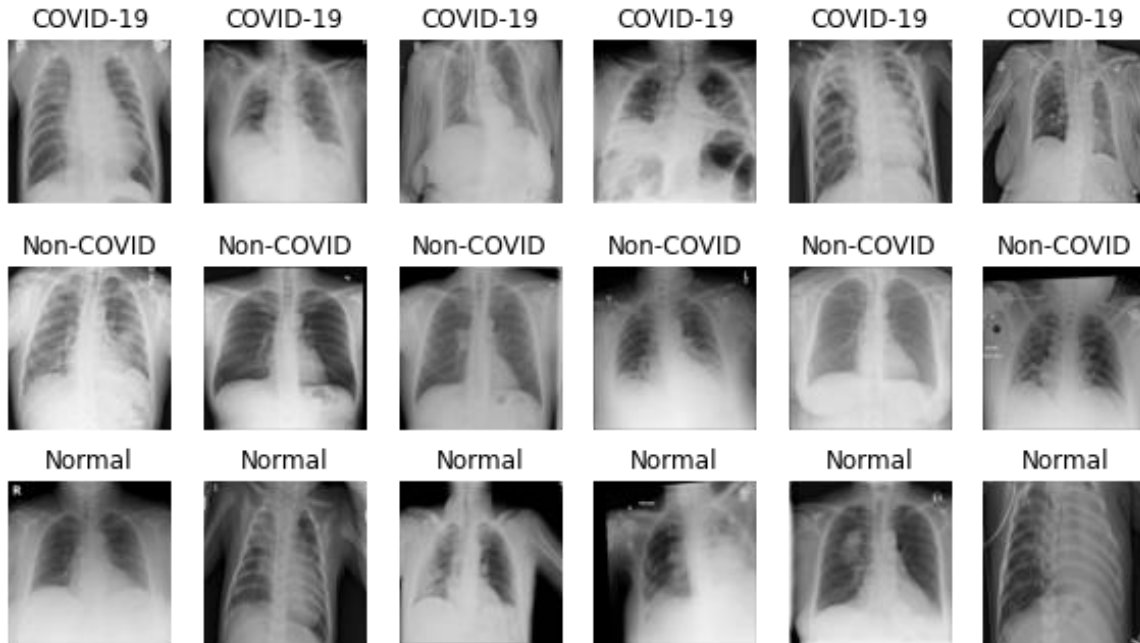
Συνοψίζοντας, στην έρευνα των (L. Wang et al., 2020) παρουσιάστηκε η πρώτη CNN αρχιτεκτονική για την αναγνώριση του κορονοϊού. Τα αποτελέσματα της έρευνας είναι αξιόπιστα και δείχνουν πως η βαθιά μάθηση μπορεί να χρησιμοποιηθεί ως εργαλείο αποφάσεων για τη διάγνωση κορονοϊού. Επίσης, η χρήση του *GSInquire* έδειξε πως η αρχιτεκτονική βασίζει τα αποτελέσματα της σε ιατρικά χαρακτηριστικά, αυξάνοντας την αξιοπιστία του συστήματος. Παράλληλα, συγκριτικά με άλλες αρχιτεκτονικές, το *COVID-Net* είναι προσπελάσιμο εξαιτίας της χαμηλής υπολογιστής πολυπλοκότητας του, όπου μπορεί να χρησιμοποιηθεί σε διάφορα συστήματα χωρίς εξειδικευμένο εξοπλισμό. Αξίζει να σημειωθεί πως το *COVID-Net* και *COVIDx* είναι ανοικτά προς το κοινό, όπου μπορεί ο κάθε ερευνητής να τα χρησιμοποιήσει. Η αρχιτεκτονική αλλά και το σετ δεδομένων είναι διαθέσιμα στο GitHub (L. Wang, 2020/2022).

## 6. Δεδομένα

Στην προκειμένη πτυχιακή εργασία, χρησιμοποιήθηκε το σετ δεδομένων από τους (Chowdhury et al., 2020; Degerli et al., 2021; Rahman et al., 2021; Tahir et al., 2021), το οποίο είναι διαθέσιμο στην πλατφόρμα *Kaggle* (*COVID-QU-Ex Dataset*, n.d.). Το εν λόγω σετ απαρτίζεται από συνολικά 33,920 αξονικές ακτινογραφίες, χωρισμένες στις παρακάτω κατηγορίες (βλ. Πίνακα 6-1).

Πίνακας 6-1: Η δομή του σετ δεδομένων COVID-QU.

#	Κατηγορία	Αριθμός εικόνων
1	COVID-19 (ακτινογραφίες με κορονοϊό)	11,956
2	Non-COVID (ιογενή ή βακτηριακή πνευμονία)	11,263
3	Normal (υγιής)	10,701



Εικόνα 6-1: Δείγμα από το σετ δεδομένων που θα χρησιμοποιηθεί στην προκυμμένη πτυχιακή εργασία.

Για τη δημιουργία του COVID-QU συλλέχθηκαν αξονικές ακτινογραφίες από ένα πλήθος αποθετηρίων. Για αυτό το λόγο, στην έρευνα των (Tahir et al., 2021) αφαιρέθηκαν διπλότυπες εικόνες, εξαιρετικά χαμηλής ποιότητας, με μεγάλη δόση ακτινοβολίας (over-exposed). Τέλος, το σετ περιέχει μόνο *posterior to anterior (PA)* και *anterior to posterior (AP)* αξονικές ακτινογραφίες. Θα ακολουθήσει ανάλυση των αποθετηρίων που χρησιμοποίησαν οι (Tahir et al., 2021) για τη δημιουργία του COVID-QU.

### 6.1. COVID-19 CXR dataset

Αρχικά, αντλήθηκαν εικόνες από το COVID-19 CXR σετ δεδομένων, το οποίο απαρτίζεται από 11,956 εικόνες με κορονοϊό. Οι εικόνες του συγκεκριμένου σετ πάρθηκαν από διάφορες πηγές. Αναλυτικά, από τις 11,956 εικόνες, οι 10,814 πάρθηκαν από το BIMCV-COVID19+ (BIMCV-COVID19 – BIMCV, n.d.), έπειτα άλλες 183 εικόνες πάρθηκαν από μία Γερμανική Ιατρική σχολή (Covid-19-Image-Repository, n.d.), στη συνέχεια 559 εικόνες πάρθηκαν από το SIRM (COVID-19 DATABASE - SIRM, n.d.), GitHub (PhD,

2020/2022), *Kaggle (COVID-19 Radiography Database*, n.d.) και *Eurorad (Eurorad.Org*, n.d.). Τέλος, 400 εικόνες πάρθηκαν από ακόμη ένα COVID-19 αποθετήριο (Haghanifar, 2020/2022).

## 6.2. RSNA CXR dataset

Στη συνέχεια, αντλήθηκαν εικόνες από το RSNA CXR (*RSNA Pneumonia Detection Challenge*, n.d.) σετ δεδομένων. Το συγκεκριμένο σετ περιέχει 26,684 εικόνες, όπου οι 8,851 είναι υγιής, 11,821 είναι αφύσικες και 6,012 είναι αμφοτερόπλευρες σκιάσεις του πνεύμονα. Από αυτό το σετ χρησιμοποιήθηκαν 8,851 υγιής εικόνες, και 6,012 αμφοτερόπλευρες σκιάσεις του πνεύμονα. Τέλος, οι 6,012 εικόνες χρησιμοποιήθηκαν στην κατηγορία *Non-COVID* του *COVID-QU* σετ.

## 6.3. Chest X-Ray Pneumonia dataset

Έπειτα, χρησιμοποιήθηκε το *Chest X-Ray Pneumonia* σετ (*Chest X-Ray Images (Pneumonia)*, n.d.). Το συγκεκριμένο σετ απαρτίζεται από 1,300 εικόνες με ιογενή πνευμονία, 1,700 εικόνες με βακτηριακή πνευμονία και 1,000 υγιής εικόνες. Εικόνες με ιογενή πνευμονία και βακτηριακή πνευμονία, ταξινομήθηκαν στην κατηγορία *Non-COVID* του *COVID-QU*.

## 6.4. PadChest dataset

Στη συνέχεια, χρησιμοποιήθηκε το PadChest (*PADCHEST – BIMCV*, n.d.) σετ δεδομένων, το οποίο περιέχει πάνω από 160,000 αξονικές ακτινογραφίες θώρακα, 67,000 ασθενών. Οι ακτινογραφίες συλλέχθηκαν από ραδιολόγους του νοσοκομείου *San Juan* στην Ισπανία από το 2009 μέχρι το 2017. Για το *COVID-QU* σετ, χρησιμοποιήθηκαν 4,000 υγιής ακτινογραφίες και 4,000 ακτινογραφίες με πνευμονία, για τις κατηγορίες *Normal* και *Non-COVID* αντίστοιχα.

## 6.5. Montgomery και Shenzhen CXR lung masks datasets

Τα συγκεκριμένα σετ δεδομένων πάρθηκαν από το νοσοκομείο Shenzhen στην Κίνα (Candemir et al., 2014) και από το Υπουργείο Υγείας και Ανθρωπίνων Υπηρεσιών της κομητείας Montgomery, Η.Π.Α (Jaeger et al., 2014). Συγκεκριμένα το σετ από Montgomery, περιέχει 80 υγιής ακτινογραφίες και 58 ακτινογραφίες με φυματίωση μαζί με τα *lung segmentation masks* της κάθε εικόνας. Το σετ από Shenzhen περιέχει 326 υγιής ακτινογραφίες και 336 ακτινογραφίες με φυματίωση. Από τις 622 συνολικά εικόνες του εν λόγω σετ, οι 566 έχουν το αντίστοιχο *lung segmentation mask*. Για το *COVID-QU* σετ, χρησιμοποιήθηκαν συνολικά 704 αξονικές ακτινογραφίες θώρακα από τα δύο σετ. Συγκεκριμένα,

πάρθηκαν όλες οι εικόνες από το Montgomery σετ και οι 566 εικόνες που έχουν το αντίστοιχο *lung segmentation* από το Shenzhen σετ. Όπως αναφέρθηκε παραπάνω, το *COVID-QU* σετ περιέχει *lung segmentation masks* για κάθε εικόνα. Για τη δημιουργία των *segmentation masks*, χρησιμοποιήθηκαν οι εικόνες από τα 2 παραπάνω.

## 6.6. QaTa-Cov19 CXR infection mask dataset

Το QaTa-Cov19 CXR infection mask dataset (Degerli et al., 2021) δημιουργήθηκε από ερευνητές των πανεπιστημίων *Qatar University* και *Tamperer University*. Απαρτίζεται από 120,000 αξονικές ακτινογραφίες, όπου οι 2,913 σχετίζονται με τον κορονοϊό. Όλες οι εικόνες έχουν και το ανάλογο *segmentation mask* αντίστοιχα.

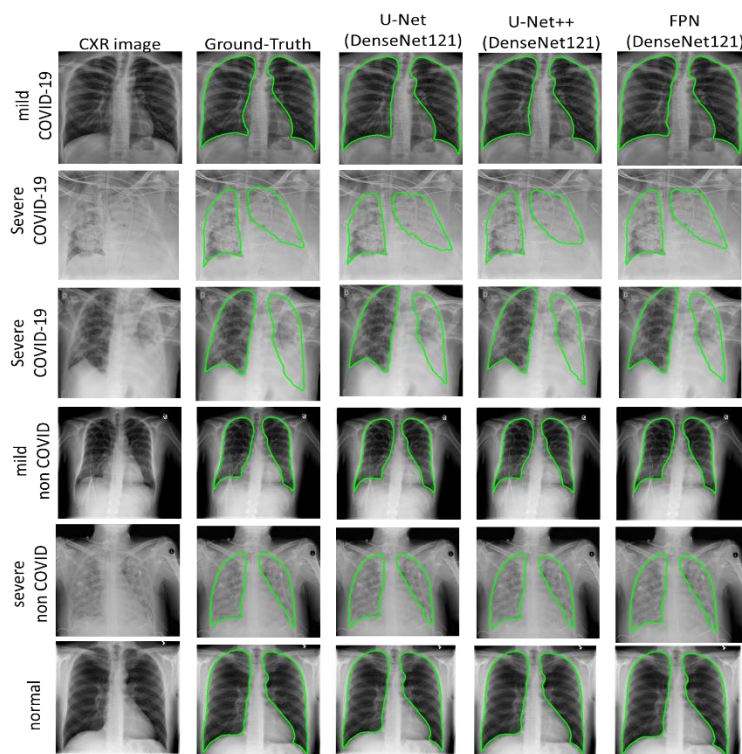
Για τη δημιουργία των *segmentation masks* του *COVID-QU* σετ χρησιμοποιήθηκε μία συνεργατική μέθοδος μεταξύ ανθρώπου και υπολογιστή. Συγκεκριμένα, η μεθοδολογία χωρίζεται σε 4 φάσεις (Tahir et al., 2021).

Στη 1<sup>η</sup> φάση, 3 παραλλαγές *U-Net* μοντέλων εκπαιδεύτηκαν πάνω στις 704 εικόνες του Montgomery και Shenzhen σετ που αναφέρθηκε παραπάνω. Το μοντέλο με την καλύτερη επίδοση χρησιμοποιείται για τη 2<sup>η</sup> φάση.

Στη 2<sup>η</sup> φάση, το μοντέλο που επιλέχθηκε προηγουμένως, εκπαιδεύεται πάνω σε ένα υποσύνολο 3000 εικόνων του *COVID-QU* σετ. Το υποσύνολο που επιλέχθηκε αντιπροσωπεύει την ποικιλότητα του γενικού σετ. Αρχικά, το μοντέλο εκπαιδεύεται σε 500 από τις 3000 εικόνες. Στη συνέχεια, τα αποτελέσματα του μοντέλου αξιολογούνται από τους ερευνητές ως “αποδεχτό”, “μη-αποδεκτό”, “αβέβαιο” και “να μη συμπεριληφθεί”. Οι μάσκες που καλύπτουν σωστά τα σημεία του πνεύμονα αξιολογούνται ως “αποδεκτές”. Οι μάσκες που δε γίνονται αποδεκτές, τροποποιούνται από τους ερευνητές και στη συνέχεια γίνονται αποδεκτές με τη σωστή μάσκα. Έπειτα, μάσκες που χαρακτηρίστηκαν ως “αβέβαιο”, είναι συνήθως σοβαρές περιπτώσεις με υψηλά επίπεδα μόλυνσης. Οι εν λόγω μάσκες αξιολογούνται από γιατρούς και μετά τροποποιούνται από τους ερευνητές με βάση τις συμβουλές των γιατρών. Τέλος, οι μάσκες που δε συμπεριλαμβάνονται στο είναι αυτές που η ποιότητα τους είναι τόσο μικρή που δεν μπορεί να χρησιμοποιηθεί. Αυτή η διαδικασία, επαναλαμβάνεται (500 εικόνες κάθε φορά) μέχρι να παραχθούν 3000 *segmentation masks* για κάθε εικόνα.

Στη 3<sup>η</sup> φάση, 6 μοντέλα βασισμένα στις αρχιτεκτονικές *U-Net*, *U-net++* και *FPN* εκπαιδεύονται πάνω στο υποσύνολο των 3000 *segmentation masks* που δημιουργήθηκε στη 2<sup>η</sup> φάση, για τη δημιουργία των *segmentation masks* του υπόλοιπου *COVID-QU* σετ (περίπου 30,900 εικόνες χωρίς μάσκες). Ως

αποτέλεσμα, συλλέχθηκαν *ground-truth masks* για 33,920 αξονικές ακτινογραφίες. Στη 4<sup>η</sup> φάση, γίνεται μία τελευταία επαλήθευση από τους γιατρούς σε 6,788 τυχαίες εικόνες. Οι εικόνες που επιλέχθηκαν τυχαία ανήκουν και στις 3 κατηγορίες του *COVID-QU* σετ.



Εικόνα 6-2: Τα αποτελέσματα της αξιολόγησης των μοντέλων όσο αφορά την παραγωγή των масκών (Tahir et al., 2021).

Στην προκειμένη πτυχιακή εργασία δε χρησιμοποιήθηκαν τα *lung segmentation masks*. Παρόλα αυτά, θεωρήθηκε σημαντικό στο να γίνει αναφορά στην εκτενή μεθοδολογία των (Tahir et al., 2021) για την παραγωγή τους. Σύμφωνα με την έρευνα που πραγματοποιήθηκε σε αυτή την πτυχιακή εργασία, βρέθηκε πως μέχρι τώρα, αυτό είναι το μεγαλύτερο σετ δεδομένων με αξονικές ακτινογραφίες θώρακα για τον κορονοϊό που περιλαμβάνει ακριβής *lung segmentation masks*.

Το *COVID-QU* σετ είναι χωρισμένο σε 3 υποσύνολα (βλ. Πίνακα 6-2), τα οποία είναι α) *Train set*, β) *Validation Set* και γ) *Test set*. Τα μοντέλα που θα χρησιμοποιηθούν στην προκειμένη πτυχιακή θα εκπαιδευτούν πάνω στο *train set*. Έπειτα, θα αξιολογηθεί η επίδοση και η ποιότητα της εκπαίδευσης τους πάνω στο *validation set*. Εφόσον τα αποτελέσματα του κάθε μοντέλου στο *validation set* είναι αποδεκτά, τότε θα γίνει μία τελική αξιολόγηση πάνω στο *test set*. Το *test set* περιέχει εικόνες που δεν έχει ξανά δει ή εκπαιδευτή το εκάστοτε μοντέλο.

**Πίνακας 6-2: Ο κατακερματισμός του σετ δεδομένων της εν λόγω πτυχιακής εργασίας.**

#	Υποσύνολο	COVID-19	Non-COVID	Normal	Συνολικά
1	Train Set	7,658 – 35%	7,208 – 33%	6,849 – 32%	21,715 – 64%
2	Validation Set	1,903 – 35%	1,802 – 33%	1,712 – 32%	5,417 – 16%
3	Test Set	2,395 – 35%	2,253 – 33%	2,140 – 32%	6,788 – 20%

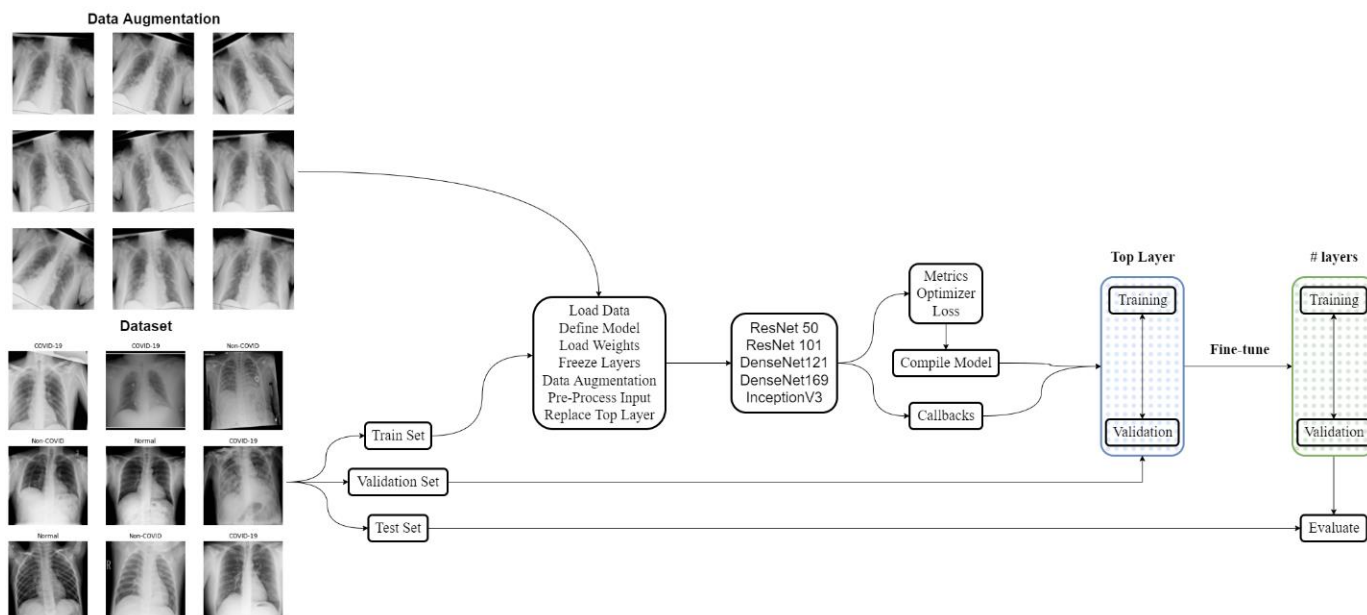
Είναι γεγονός πως, όσο αφορά το θέμα της συγκεκριμένης πτυχιακής, δεν υπάρχουν αρκετά σετ δεδομένων που περιέχουν ξεχωριστό *Test set*. Σχεδόν σε όλες τις έρευνες που εξετάστηκαν, τα δεδομένα που χρησιμοποιούσαν ήταν χωρισμένα σε 2 υποσύνολα και όχι 3. Αδιαμφισβήτητα, ένα 3<sup>ο</sup> σετ, το οποίο δεν έχει χρησιμοποιηθεί για εκπαίδευση, προσφέρει στον ερευνητή καλύτερη επίγνωση πάνω στην επίδοση του μοντέλου που αξιολογείται.

Επίσης, από τον Πίνακα 6-2 μπορεί να παρατηρηθεί πως το σετ δεδομένων είναι σωστά ισορροπημένο, όσο αφορά τον αριθμό εικόνων σε κάθε κατηγορία και υποσύνολο. Συγκεκριμένα, οι εικόνες που ανήκουν στο *train set* αντιστοιχούν στο 64% του συνόλου, και στη συνέχεια, οι εικόνες του *validation set* και *train set* αντιστοιχούν στο 16% και 20% αντίστοιχα. Παράλληλα όσο αφορά των αριθμό εικόνων σε κάθε κατηγορία και στα 3 υποσύνολα, μπορεί να παρατηρηθεί πως οι εικόνες είναι χωρισμένες ισάξια όπου κάθε κατηγορία καταλαμβάνει περίπου το 35% του συνολικού μεγέθους.

Στην έρευνα των (Maguolo & Nanni, 2020) αποδείχθηκε πως μοντέλα βαθιάς μάθησης έχουν την ικανότητα να μάθουν την πηγή της κάθε εικόνας. Στην έρευνα των (Tahir et al., 2021) δεν έγινε κάποια αξιολόγηση ούτε σχολιασμός πάνω σε αυτό. Ωστόσο, το σετ δεδομένων είναι επαρκές ως προς το μέγεθος του, και οι πηγές που χρησιμοποίησαν είναι αρκετές έτσι ώστε πιθανότατα να μην παρουσιαστεί το εν λόγω πρόβλημα.

## 7. Μεθοδολογία

Σε αυτό το κεφάλαιο θα γίνει σχολιασμός της μεθοδολογίας που χρησιμοποιήθηκε στην προκειμένη πτυχιακή εργασία για την ανίχνευση κορονοϊού σε αξονικές ακτινογραφίες (βλ. Εικόνα 7-1).



Εικόνα 7-1: Συνοπτικά η μεθοδολογία της εν λόγω πτυχιακής εργασίας.

### 7.1. Αρχιτεκτονικές και μοντέλα βαθιάς μάθησης

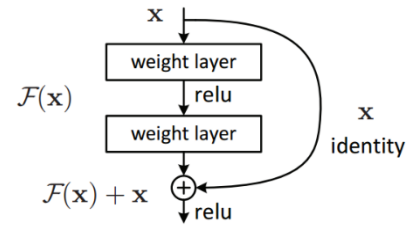
Στην πρώτη φάση, έγινε έρευνα για την επιλογή του κάθε μοντέλου που θα χρησιμοποιηθεί για την πρόβλεψη κορονοϊού. Μέσα από διάφορα ερευνητικά άρθρα, παρατηρήθηκε μεγάλη χρήση των αρχιτεκτονικών ResNet (He et al., 2015b), DenseNet (G. Huang et al., 2018) και Inception (Szegedy et al., 2015). Για αυτό το λόγο, επιλέχθηκαν τα μοντέλα α) ResNet50, β) ResNet101, γ) DenseNet121, δ) DenseNet169 και ε) InceptionV3. Θα ακολουθήσει μία μικρή περίληψη για την κάθε αρχιτεκτονική παρακάτω.

#### 7.1.1. ResNet

Στην έρευνα των (He et al., 2015b) παρουσιάστηκε η αρχιτεκτονική Residual Network – ResNet με σκοπό την επίλυση του προβλήματος vanishing/exploding gradients (Bengio et al., 1994). Η εν λόγω αρχιτεκτονική απαρτίζεται κυρίως από residual blocks και batch normalization layers - BN. Συγκεκριμένα,



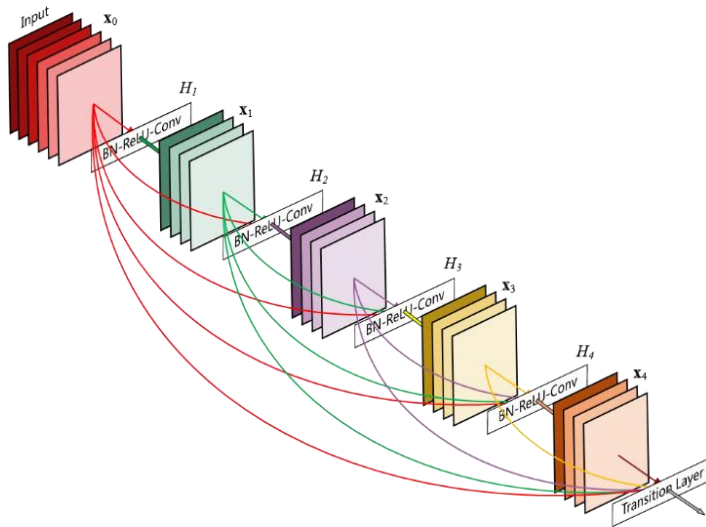
κάθε residual block (βλ. Εικόνα 7-2) αποτελείται από convolutional layers και shortcut connections, όπου τα *shortcut connections*, μέσω πρόσθεσης, επιτρέπουν τη συγχώνευση της εισόδου (input  $X$ ) μαζί με την έξοδο (output  $f(x)$ ) όπου το αποτέλεσμα είναι  $f(x) + X$ . Τέλος, υπάρχουν διάφορες εναλλαγές της αρχιτεκτονικής σχετικά με τον αριθμό επιπέδων που απαρτίζεται, όπως ResNet18, ResNet34, ResNet50, ResNet101 και ResNet152, όπου το κάθε μοντέλο περιέχει 18, 34, 50, 101 και 152 layers αντίστοιχα.



Εικόνα 7-2: Ένα Residual Block της αρχιτεκτονικής ResNet (He et al., 2015).

Επίσης, η αρχιτεκτονική έχει εκπαιδευτεί πάνω στο σετ δεδομένων *ImageNet* το οποίο περιέχει 1000 κατηγορίες ταξινόμησης και πάνω από 1,000,000 εικόνες. Όπως αναφέρθηκε, στην προκειμένη έρευνα θα χρησιμοποιηθεί το μοντέλο *ResNet50* και *ResNet101*.

### 7.1.2. DenseNet



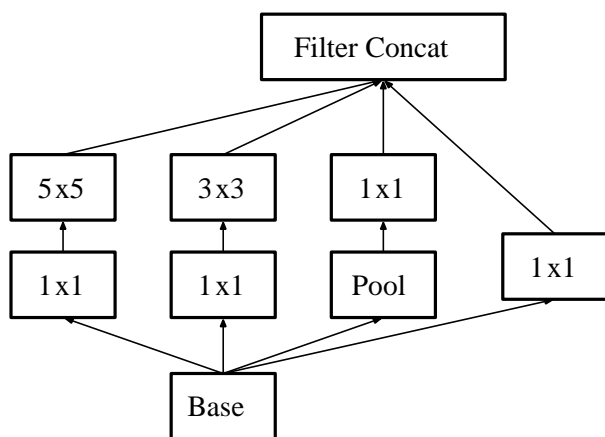
Εικόνα 7-3: Ένα DenseBlock με 5 layers (Huang et al., 2018).

Η αρχιτεκτονική *DenseNet* – *Dense Convolutional Network* δημιουργήθηκε από τους (G. Huang et al., 2018) όπου το κάθε layer συνδέεται με τα υπόλοιπα layers. Γενικότερα, κλασικά *convolutional networks* με  $L$  layers, έχουν  $L$  συνδέσεις (μία σε κάθε layer), στην αρχιτεκτονική *DenseNet* υπάρχουν  $\frac{L(L+1)}{2}$  απευθείας συνδέσεις.

Για κάθε επίπεδο (layer), τα *feature maps* όλων των προηγούμενων επιπέδων μαζί με τα *feature maps* του εν λόγω επιπέδου χρησιμοποιούνται ως είσοδο στα επόμενα επίπεδα (βλ. Εικόνα 7-3). Μερικά πλεονεκτήματα της συγκεκριμένης αρχιτεκτονικής είναι πως μειώνουν το πρόβλημα των *vanishing gradients*, ενδυναμώνουν τη ροή της πληροφορίας και υπάρχει σημαντική μείωση στον αριθμό των παραμέτρων.

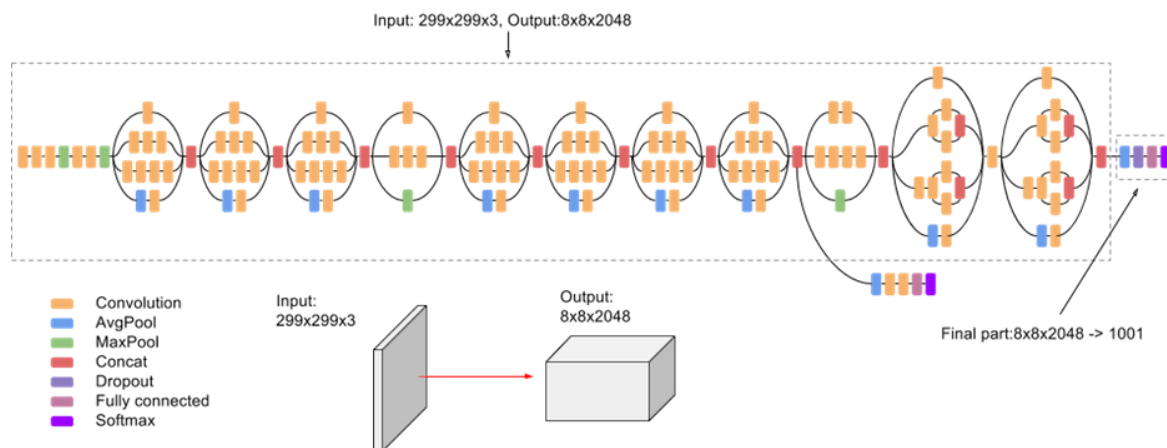
### 7.1.3. Inception Network

Η αρχιτεκτονική *Inception* παρουσιάστηκε από τους (Szegedy et al., 2015) και απαρτίζεται κυρίως από *Inception modules*. Τα δομικά στοιχεία των συγκεκριμένων *modules* είναι *1x1 Convolutions* τα οποία παρουσιάστηκαν από τους (M. Lin et al., 2014) με το όνομα «Network In Network». Σκοπός τους είναι να μειώσουν τις διαστάσεις των εισερχόμενων δεδομένων. Στη συνέχεια, υπάρχουν τα *3x3* και *5x5 convolutions* όπου επιτρέπουν στην αρχιτεκτονική να μάθει μοτίβα και χαρακτηριστικά που υπάρχουν στην εικόνα, σε διαφορετικές κλίμακες. Πριν τη δημιουργία της εν λόγω αρχιτεκτονικής, οι ερευνητές έπρεπε να αποφασίσουν και να επιλέξουν το μέγεθος των *convolutions*. Στην προκειμένη περίπτωση δε χρειάζεται διότι το *Inception Net* συμπεριλαμβάνει και τα 3 μεγέθη. Τα τελευταία δομικά στοιχεία του *Inception module* είναι το *Max Pooling layer* και το *Concatenation layer*. Η παρακάτω εικόνα απεικονίζει το αρχικό σχέδιο των *Inception modules* που παρουσιάστηκε από τους (Szegedy et al., 2014).



Εικόνα 7-4: Ο αρχικός σχεδιασμός του *Inception Module* (Szegedy et al., 2014).

Στην προκειμένη πτυχιακή εργασία θα χρησιμοποιηθεί η αρχιτεκτονική *Inception-v3* η οποία φέρνει μερικές βελτιώσεις συγκριτικά με τη 1<sup>η</sup> έκδοση. Συγκεκριμένα, στη 3<sup>η</sup> έκδοση της αρχιτεκτονικής υπάρχει η μέθοδος *Label Smoothing* όπου εισάγει “θόρυβο” ή “noise” στις κατηγορίες (Goodfellow et al., 2016). Στη συνέχεια, υπάρχουν *Factorized 7x7 Convolutions*, βοηθητικοί ταξινομητές (*auxiliary classifiers*), και η χρήση των *Batch Normalization layers*. Στην παρακάτω εικόνα φαίνεται ολοκληρωμένα η δομή της αρχιτεκτονικής.



Εικόνα 7-5: Το διάγραμμα της *Inception-v3* αρχιτεκτονικής (Advanced Guide to Inception v3 | Cloud TPU, n.d.).

Με την εύρεση των αρχιτεκτονικών, μπορεί να ξεκινήσει η επόμενη φάση της μεθοδολογίας, η οποία είναι το σετ δεδομένων, η επεξεργασία του και τα εργαλεία που θα χρησιμοποιηθούν για την εκπαίδευση του κάθε μοντέλου.

## 7.2. Εργαλεία

Όσο αφορά το προγραμματιστικό κομμάτι της πτυχιακής, επιλέχθηκε το εργαλείο Visual Studio Code της Microsoft ή VS Code (Visual Studio Code - Code Editing. Redefined, n.d.) στην έκδοση 1.70.2. Το συγκεκριμένο εργαλείο προσφέρει αποσφαλμάτωση του κώδικα (debugging), υπογράμμιση σύνταξης (syntax highlighting), έξυπνη ολοκλήρωση κώδικα, επανασχεδιασμό του προγράμματος (code refactoring), και ενσωματωμένο Git. Ακόμη ένα προτέρημα του VS Code είναι τα διαθέσιμα εργαλεία (extensions) που δίνει στον χρήστη. Τα εν λόγω εργαλεία αυξάνουν την χρηστικότητα του προγράμματος εφόσον ο κάθε προγραμματιστής μπορεί να εγκαταστήσει τα απαραίτητα εργαλεία που χρειάζεται για την υλοποίηση του προγράμματός του. Τα σημαντικά εργαλεία που εγκαταστάθηκαν για την προκειμένη πτυχιακή είναι 1) Jupyter Notebook, 2) Python, 3) Pylance. Παράλληλα, χρησιμοποιήθηκαν τα εργαλεία draw.io (Diagrams.Net, n.d.) και Excel (Microsoft Excel, n.d.) για τη δημιουργία διαγραμμάτων και γραφημάτων. Επίσης, όσο αφορά τη συγγραφή της πτυχιακής εργασίας, χρησιμοποιήθηκε το εργαλείο Word (Microsoft Word, n.d.). Τέλος, χρησιμοποιήθηκε η πλατφόρμα GitHub (Mariosconsta/Thesis, n.d.) όπου είναι ανεβασμένος ο πηγαίος κώδικας της εργασίας.

Όσο αφορά τη γλώσσα προγραμματισμού, χρησιμοποιήθηκε η Python στην έκδοση 3.10.2 (Python, n.d.). Η Python είναι μία αντικειμενοστραφής γλώσσα προγραμματισμού υψηλού επιπέδου, με ένα τεράστιο αριθμό από διαθέσιμες βιβλιοθήκες (libraries) για κάθε κλάδο όπως, στατιστική ανάλυση, δημιουργία γραφικής διεπαφής χρήστη – Graphical User Interface, βάσεις δεδομένων, οπτικοποίηση δεδομένων και επεξεργασία δεδομένων.

### 7.3. Βιβλιοθήκες

Στη προκειμένη πτυχιακή εργασία, οι βασικές βιβλιοθήκες που χρησιμοποιήθηκαν είναι NumPy (NumPy, n.d.), Tensorflow (TensorFlow, n.d.), Keras (Keras, n.d.), Scikit-Learn (Scikit-Learn, n.d.), Pandas (Pandas, n.d.), Matplotlib (Matplotlib, n.d.), Jupyter Notebook (Project Jupyter, n.d.) και Seaborn (Seaborn, n.d.). Παρακάτω θα ακολουθήσει μία μικρή περίληψη της κάθε βιβλιοθήκης.

#### 7.3.1. NumPy

Η βιβλιοθήκη NumPy (NumPy, n.d.) δημιουργήθηκε το 2005 από τον Travis Oliphant και είναι το θεμελιώδες μαθηματικό εργαλείο για επεξεργασία πινάκων στη γλώσσα Python. Η βιβλιοθήκη είναι ανοικτό προς το κοινό (open-source), όπου μπορεί να χρησιμοποιηθεί ελεύθερα και δωρεάν. Συγκεκριμένα, η βιβλιοθήκη παρέχει στον χρήστη αρκετά μαθηματικά εργαλεία για επεξεργασία πολυδιάστατων πινάκων και παράλληλα μία ποικιλία από διαδικασίες που μπορούν να εφαρμοστούν στους εν λόγω πίνακες. Μερικά από αυτά είναι, μαθηματικές και λογικές πράξεις, αλλαγή σχήματος, επιλογή και ταξινόμηση πινάκων, εφαρμογή μετασχηματισμών Fourier, απλές πράξεις γραμμικής άλγεβρας, και απλές στατιστικές μαθηματικές διαδικασίες.

#### 7.3.2. Scikit-Learn

Το Scikit-Learn (Scikit-Learn, n.d.) είναι μία από της πιο διαδεδομένες βιβλιοθήκες όσο αφορά τη χρήση μηχανικής μάθησης. Δημιουργήθηκε από τον David Cournapeau το 2007 και έπειτα το 2010 οι Fabian Pedregosa, Gael Varoquaux, Alexandre Gramfort, and Vincent Michel ανέλαβαν την υλοποίηση της βιβλιοθήκης όπου στη 1<sup>η</sup> Φεβρουαρίου, 2010 έγινε διαθέσιμη η πρώτη της έκδοση. Η βιβλιοθήκη γράφτηκε κυρίως σε Python και έχει ως βάση τη βιβλιοθήκη NumPy, SciPy και Matplotlib.

Συγκεκριμένα, η βιβλιοθήκη παρέχει μία μεγάλη ποικιλία από εργαλεία για μηχανική μάθηση και στατιστική μοντελοποίηση όπως *classification*, *regression*, *clustering*, και *dimensionality reduction*. Όπως αναφέρθηκε, το *Scikit-Learn* παρέχει ένα μεγάλο αριθμό εργαλείων. Σε αυτή την πτυχιακή εργασία χρησιμοποιήθηκε το ***classification report*** (*Sklearn Metrics Classification Report*, n.d.) και ***confusion matrix*** (*Sklearn Metrics Confusion Matrix*, n.d.). Θα ακολουθήσει αργότερα στο παρόν κεφάλαιο μία πιο αναλυτική περιγραφή των μετρήσεων που χρησιμοποιήθηκαν για την αξιολόγηση των μοντέλων της πτυχιακής εργασίας.

#### 7.3.3. Pandas

Το Pandas (Pandas, n.d.) είναι μία βιβλιοθήκη που χρησιμοποιείται κυρίως για ανάλυση δεδομένων σε συνδυασμό με μηχανική μάθηση. Παράλληλα, η εν λόγω βιβλιοθήκη βασίζεται πάνω στη NumPy που αναφέρθηκε παραπάνω. Για τον λόγο αυτό, μπορεί να εφαρμόσει συγχώνευση ή ένωση πινάκων. Η βιβλιοθήκη παρέχει ποικίλα εργαλεία για ανάλυση δεδομένων. Μερικά από αυτά είναι καθαρισμός, γέμισμα, κανονικοποίηση και απεικόνιση δεδομένων, έπειτα προσφέρει τρόπους για στατιστική ανάλυση και, επιθεώρηση δεδομένων.

#### 7.3.4. Matplotlib

Το Matplotlib (Matplotlib, n.d.) δημιουργήθηκε από τον John D. Hunter και συνεχίστηκε από τον Michael Droettboom και Thomas Caswell. Η βιβλιοθήκη προσφέρει ένα API για ενσωμάτωση και δημιουργία γραφικών παραστάσεων σε εφαρμογές όπως JupyterLab. Μπορεί να παρατηρηθεί πως το Matplotlib συνδυάζεται συχνά με τη μηχανική μάθηση για την δημιουργία γραφικών παραστάσεων όσο αφορά τις διάφορες μετρήσεις του μοντέλου ή για απεικόνιση δεδομένων.

#### 7.3.5. Seaborn

Το Seaborn (Seaborn, n.d.) είναι ένα εργαλείο απεικόνισης δεδομένων βασισμένο πάνω στο Matplotlib. Μερικές λειτουργίες του Seaborn είναι η επισήμανση γραφικών παραστάσεων γραμμικής άλγεβρας και απεικόνιση κατανομών univariate ή bivariate. Επίσης, προσφέρει ένα API για την εύρεση σχέσεων μεταξύ μεταβλητών. Με τη χρήση του Seaborn μπορούν να δημιουργηθούν διάφορα γραφήματα (βλ. Εικόνα 2). Μερικά από αυτά είναι, distribution plots, κυκλικά διαγράμματα (pie chart), γραφήματα ράβδων (bar chart), διαγράμματα διασποράς (scatter plots) και χάρτες θερμότητας (heat map).

#### 7.3.6. Jupyter Notebook

Το Jupyter Notebook (Project Jupyter, n.d.) είναι μία εφαρμογή ανοικτού κώδικα για τη δημιουργία και κατανομή αρχείων που περιέχουν κώδικα, εξισώσεις, απεικονίσεις και κείμενο. Το όνομα προέρχεται από της γλώσσες που υποστηρίζει το εργαλείο οι οποίες είναι Julia, Python και R.

#### 7.3.7. Tensorflow

Το Tensorflow (TensorFlow, n.d.) είναι μία βιβλιοθήκη ανοικτού κώδικα για μηχανική μάθηση η οποία δημιουργήθηκε από την ομάδα Google Brain το 2015. Οι χρήσεις της εν λόγω βιβλιοθήκης είναι ποικίλες, ωστόσο χρησιμοποιείτε κυρίως για την εκπαίδευση μοντέλων βαθιάς μάθησης. Επίσης, η βιβλιοθήκη μπορεί να χρησιμοποιηθεί σε διάφορες γλώσσες προγραμματισμού όπως Python, Javascript, C++ και Java.

Μερικά χαρακτηριστικά του Tensorflow είναι Auto Differentiation, Eager execution, Distribute, Losses, Metrics και Optimizers. Επίσης, είναι πλήρως ενσωματωμένο με NumPy και μπορεί να χρησιμοποιηθεί σε συνδυασμό με άλλες βιβλιοθήκες ή επεκτάσεις όπως Tensorflow Graphics, Tensorflow Federated, Tensorflow Cloud, Tensorflow Model Optimization, Probability, Quantum και Decision Forests. Τέλος, η Google έχει δημιουργήσει ένα Tensorflow Jupyter Notebook το οποίο μπορεί να χρησιμοποιηθεί στην πλατφόρμα Google Collab χωρίς κάποια εγκατάσταση.

#### 7.3.8. Keras

Το Keras (Keras, n.d.) είναι ένα high-level API που λειτουργεί ως διεπαφή για τη βιβλιοθήκη Tensorflow. Συγκεκριμένα, η εν λόγω βιβλιοθήκη (Keras) δημιουργήθηκε με στόχο την εύκολη χρήση και επεκτασιμότητα εφαρμογών βαθιάς μηχανικής μάθησης. Πέρα από αυτό, το βασικό χαρακτηριστικό της βιβλιοθήκης είναι τα αυτόνομα δομοστοιχεία (standalone modules) που μπορούν να συνδυασθούν για την εύκολη και απλή δημιουργία νέων μοντέλων βαθιάς μάθησης, όπως νευρωνικά επίπεδα (neural layers), συναρτήσεις κόστους (cost functions), σταθεροποιητές (optimizers), συστήματα αρχικοποίησης (initialization schemes), συναρτήσεις ενεργοποίησης (activation functions), και συστήματα τακτοποίησης (regularization schemes).

### 7.4. Προετοιμασία και προ-επεξεργασία δεδομένων

#### 7.4.1. Προετοιμασία δεδομένων

Όπως αναφέρθηκε στο κεφάλαιο «Δεδομένα», μέσα από εκτενή έρευνα, επιλέχθηκε το σετ δεδομένων από τους (Degerli et al., 2021; Rahman et al., 2021; Tahir et al., 2021) το οποίο είναι διαθέσιμο στην πλατφόρμα Kaggle (COVID-QU-Ex Dataset, n.d.). Το συγκεκριμένο σετ επιλέχθηκε για 3 βασικούς λόγους. 1) Το μέγεθος. Μέσα από την έρευνα που έγινε στην προκειμένη πτυχιακή, αυτό είναι το μεγαλύτερο σετ δεδομένων το οποίο σχετίζεται με αξονικές ακτινογραφίες σε κορονοϊό που βρέθηκε μέχρι τώρα. 2) Ο διαθέσιμος κατακερματισμός του σετ σε *train*, *validation* και *test* που συμπεριλαμβάνεται. Όπως αναφέρθηκε, είναι σημαντικό να υπάρχει ένα υποσύνολο του σετ δεδομένων, το οποίο είναι άγνωστο προς το μοντέλο για την τελευταία αξιολόγηση (*test set*). 3) Η συμπερίληψη των *lung segmentation masks*. Παρόλο που δεν υλοποιήθηκαν στη συγκεκριμένη φάση της πτυχιακής εργασίας, οι μάσκες που υπάρχουν στο σετ για κάθε εικόνα είναι τεράστιο προτέρημα. Στόχος για το μέλλον, είναι η βελτίωση της πτυχιακής έρευνας, όπου θα γίνει υλοποίηση των *segmentation masks*, για εντοπισμό του κορονοϊού. Επιπρόσθετα, όπως αναφέρθηκε στο κεφάλαιο “Δεδομένα” το σετ είναι ισορροπημένο όσο αφορά τον αριθμό εικόνων

σε κάθε υποσύνολο και κατηγορία. Για τον λόγο αυτό, δεν εφαρμόστηκε κάποια μέθοδος για την κατανομή (*distribution*) των εικόνων.

Έπειτα, πριν τη φάση του *Data Augmentation*, χρειάστηκε να εφαρμοσθούν μερικά βήματα. Το 1<sup>ο</sup> είναι το μέγεθος των εικόνων. Κάθε αρχιτεκτονική είναι συμβατή με ένα συγκεκριμένο μέγεθος εικόνων. Για αυτό τον λόγο, είναι σημαντικό να μελετηθεί η κάθε αρχιτεκτονική και έπειτα να γίνουν οι απαραίτητες αλλαγές. Στην εν λόγω πτυχιακή εργασία, το αρχικό μέγεθος των εικόνων ήταν (256,256,3) και όπως αναφέρθηκε παραπάνω, χρησιμοποιήθηκαν οι αρχιτεκτονικές *ResNet50*, *ResNet101*, *DenseNet121*, *DenseNet169* και *InceptionV3*. Οι *ResNet* και *DenseNet* χρειάζονται (224,224,3) ως μέγεθος εικόνας, και αντίστοιχα η *InceptionV3* χρειάζεται κατά ελάχιστον (75,75,3) (*InceptionV3*, n.d.), όπου το μέγεθος καθορίζεται ως (ύψος, πλάτος, βάθος) και το βάθος είναι ο αριθμός των καναλιών (*channels*) της εικόνας. Η μέθοδος παρεμβολής (*interpolation*) που χρησιμοποιήθηκε για την αλλαγή μεγέθους είναι διγραμμική (*bilinear*). Συνεπώς, για τα μοντέλα που βασίζονται στις αρχιτεκτονικές *ResNet* και *DenseNet* το μέγεθος των εικόνων άλλαξε σε (224,224,3). Όσο αφορά την *InceptionV3* το μέγεθος δεν άλλαξε από το αρχικό (256,256,3). Έπειτα, το 2<sup>ο</sup> βήμα είναι ο κατακερματισμός των υποσυνόλων του σετ δεδομένων (*train*, *validation* και *test*) σε δεσμίδες (*batches*) όπου το μέγεθος της κάθε δεσμίδας είναι 16 (*batch size*). Το 3<sup>ο</sup> βήμα είναι η προετοιμασία των ετικετών (*labels*) της κάθε εικόνας. Ουσιαστικά οι ετικέτες είναι τα ονόματα των κλάσεων που αντιστοιχούν σε κάθε εικόνα, όπου η εικόνα *X* ανήκει στην κλάση *Y*. Συγκεκριμένα, οι ετικέτες είναι κωδικοποιημένες κατηγορηματικά (*categorical encoding*) όπου κάθε ετικέτα είναι ένας *float32* τανυστής (*tensor*) της μορφής (*batch\_size*, *num\_classes*) όπου αντιπροσωπεύει *one-hot encoding* του ευρετηρίου των κλάσεων (*classes*).

Τα βήματα που αναλύθηκαν παραπάνω υλοποιήθηκαν με τη λειτουργία *image\_dataset\_from\_directory()* (*Tf.Keras.Utils.Image\_dataset\_from\_directory*, n.d.) του *Tensorflow* και *Keras*, όπου επιστρέφει ένα *tf.Data.Dataset* αντικείμενο, το οποίο είναι μία πλειάδα (*images*, *labels*), όπου τα *images* έχουν το σχήμα (*batch\_size*, *image\_size[0]*, *image\_size[1]*, *num\_channels*) και τα *labels* έχουν το σχήμα (*batch\_size*, *num\_classes*). Για παράδειγμα, στην περίπτωση του *InceptionV3* το αντικείμενο που θα επιστρέψει η εν λόγω λειτουργία θα έχει το σχήμα α) (16, 256, 256, 3) το οποίο είναι το *images* και β) είναι η ετικέτες (16, 3).

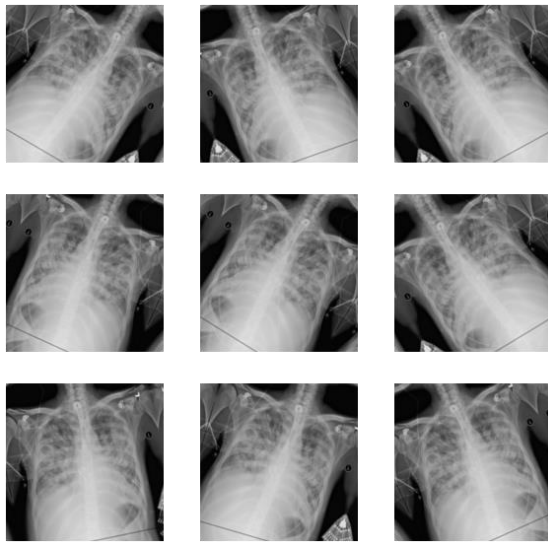
#### 7.4.2. Προ-επεξεργασία δεδομένων – *Data Augmentation*

Βάση των (Halevy et al., 2009; Sun et al., 2017), είναι γενικά αποδεκτό πως μεγαλύτερα σετ δεδομένων συνεπάγονται σε καλύτερα βαθιά μοντέλα πρόβλεψης σχετικά με την απόδοση τους. Όσο αφορά τη χρήση της βαθιάς μάθησης (*Deep Learning*) σε συνδυασμό με μηχανική όραση (*computer vision*) στον τομέα της Ιατρικής, αρκετές από τις εικόνες που χρησιμοποιούνται για την εκπαίδευση των μοντέλων προέρχονται

από αξονικές τομογραφίες (*CT Scans*), μαγνητικές τομογραφίες (*MRI*), και αξονικές ακτινογραφίες (*X-Ray*). Η συλλογή τέτοιου είδους εικόνων κοστίζει, απαιτεί χρόνο, κόπο και εξειδικευμένο προσωπικό. Παράλληλα, είναι ακόμη πιο δύσκολη η δημιουργία μεγάλων ιατρικών σετ δεδομένων εφόσον πρέπει να ληφθούν υπόψη τα προσωπικά δεδομένα του ασθενή, η αναγκαιότητα για επαγγελματίες του χώρου όσο αφορά την τιτλοφόρηση της κάθε εικόνας, και το εργατικό κόστος που απαιτείται (Shorten & Khoshgoftaar, 2019).

Μέσα από τη βιβλιογραφική ανασκόπηση που έγινε στην προκειμένη πτυχιακή εργασία, φαίνεται πως αρκετά σετ δεδομένων είναι μικρά, και αυτό οφείλεται στους λόγους που αναφέρθηκαν παραπάνω, αλλά και στο γεγονός του ότι ο κορονοϊός είναι μία σχετικά νέα ασθένεια. Ως αποτέλεσμα, η εκπαίδευση μοντέλων βαθιάς μάθησης μπορεί να οδηγήσει στην υπερπροσαρμογή (*overfitting*) του *training* σετ όταν τα δεδομένα δεν είναι επαρκές, και αυτό οφείλεται στη μεγάλη υπολογιστή χωρητικότητα του κάθε μοντέλου. *Overfitting* είναι όταν ένα μοντέλο έχει πολύ καλή απόδοση στο *training* σετ αλλά καθόλου καλή στο *test* σετ. Αυτό το φαινόμενο υποδηλώνει πως το μοντέλο δεν μπορεί να χρησιμοποιηθεί σε δεδομένα

που δεν έχει ξαναδεί διότι η απόδοση στα εν λόγω δεδομένα είναι χαμηλή.



**Εικόνα 7-6: Παράδειγμα εφαρμογής *Data Augmentation* 9 φορές σε 1 εικόνα του σετ δεδομένων της προκυμμένης πτυχιακής εργασίας**

Επιπρόσθετα, μερικές φορές παρουσιάζεται το πρόβλημα της δυσαναλογίας στα σετ δεδομένων, όπου ο αριθμός των δειγμάτων σε κάθε κατηγορία είναι μονομερής. Πρακτικά, αυτό σημαίνει πως όταν ένα σετ περιέχει πολύ περισσότερα δεδομένα στη μία κατηγορία και πολύ λιγότερα στην άλλη, τότε ενδέχεται το μοντέλο να μάθει πολύ καλά να ξεχωρίζει και να αναγνωρίζει δεδομένα που ανήκουν στην κατηγορία με τα περισσότερα δεδομένα, αλλά όχι τόσο καλά όσο αφορά την κατηγορία με τα λιγότερα.

Η μέθοδος του *Data Augmentation* βοηθάει στην αντιμετώπιση των προβλημάτων που αναφέρθηκαν παραπάνω αυξάνοντας τεχνητά το μέγεθος του *train* σετ με αποτέλεσμα να υπάρχει περισσότερη πληροφορία διαθέσιμη προς το μοντέλο. Επιπλέον, μπορεί να εφαρμοσθεί *Data Augmentation* σε συγκεκριμένες κατηγορίες του σετ, πράγμα που διευκολύνει στην αντιμετώπιση της δυσαναλογίας.

Υπάρχουν ποικίλοι μετασχηματισμοί που μπορούν να εφαρμοσθούν στις εικόνες μέσω *Data Augmentation*, μερικοί από αυτούς είναι α) γεωμετρικοί, β) χρωματικού χώρου, και δ) ανάμειξη εικόνων. Είναι σημαντικό



να ληφθεί υπόψιν η ασφάλεια χρήσης του κάθε μετασχηματισμού. Για παράδειγμα, γεωμετρικοί μετασχηματισμοί όπως τυχαία περιστροφή και αναστροφή είναι ασφαλές σε εικόνες με σκύλους και γάτες αλλά όχι τόσο σε εικόνες με ψηφία όπου το 6 μπορεί να μετατραπεί σε 9. Για αυτό το λόγο, στην προκειμένη πτυχιακή θα εφαρμοσθούν μόνο γεωμετρικοί μετασχηματισμοί στις εικόνες του *train set* μέσω του *Tensorflow*, οι οποίοι είναι, **1) τυχαία περιστροφή ( $\pm 10^\circ$ )** και **2) τυχαία οριζόντια αναστροφή**. Οι παραπάνω μετασχηματισμοί εφαρμόστηκαν με τη χρήση των μεθόδων *RandomFlip()* (*Tf.Keras.Layers.RandomFlip*, n.d.) και *RandomRotation()* (*Tf.Keras.Layers.RandomRotation*, n.d.) του *Tensorflow* και *Keras*. Όσο αφορά την τυχαία περιστροφή, η μέθοδος παρεμβολής που χρησιμοποιήθηκε ήταν η διγραμμική (*bilinear*).

Υπάρχουν 2 βασικοί τρόποι για το πως μπορεί να εφαρμοσθεί το *Data Augmentation* στο σετ δεδομένων. Ο 1<sup>ος</sup> τρόπος είναι να γίνει προ επεξεργασία πριν τη φάση του μοντέλου, κατευθείαν πάνω στο σετ δεδομένων, έτσι ώστε όταν περάσει στο μοντέλο, τα δεδομένα θα είναι ήδη προ επεξεργασμένα. Ο 2<sup>ος</sup> τρόπος είναι να εφαρμοσθεί *Data Augmentation* κατά τη διάρκεια της εκτέλεσης του μοντέλου.

Στην προκειμένη πτυχιακή επιλέχθηκε ο 2<sup>ος</sup> τρόπος για τους παρακάτω λόγους. 1) Η προ επεξεργασία συμβαίνει ταυτόχρονα με την εκπαίδευση του μοντέλου. Ιδιαίτερα η εν λόγω μέθοδος είναι πιο αποδοτική με τη χρήση κάρτας γραφικών (*GPU*). 2) Το μοντέλο γίνεται *portable* και μειώνεται η ασυμμετρία της εκπαίδευσης. Αυτό σημαίνει πως το μοντέλο μπορεί να χρησιμοποιηθεί από άλλους χωρίς να χρειάζονται περισσότερες πληροφορίες για το πως έγιναν *encode* και *normalized* τα δεδομένα, διότι συμβαίνουν μέσα στο μοντέλο. 3) Για τον ίδιο λόγο, το μοντέλο θα μπορεί να τρέξει σε ανεπεξέργαστα δεδομένα χωρίς να πρέπει ο χρήστης να κάνει κάποια προ επεξεργασία από πριν. Μετά την ολοκλήρωση της προ επεξεργασίας του σετ δεδομένων, ακολουθεί ο προσδιορισμός του μοντέλου και η μέθοδος *Transfer Learning*.

## 7.5. Προσδιορισμός μοντέλου και *Transfer Learning*

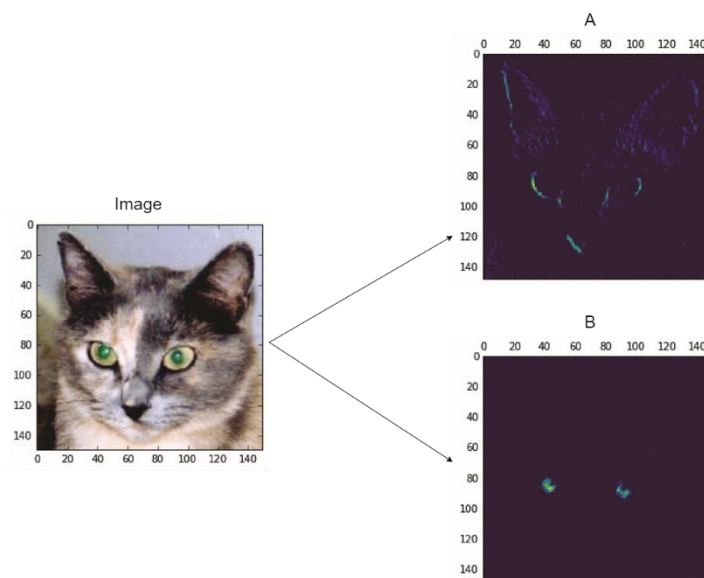
Σε αυτό το κεφάλαιο θα αναλυθεί η μέθοδος προσδιορισμού του εκάστοτε μοντέλου. Παράλληλα θα γίνει επεξήγηση της μεθόδου *Transfer Learning*.

### 7.5.1. Transfer Learning

Η εν λόγω μέθοδος χρησιμοποιείται με σκοπό τη βελτίωση της εκμάθησης ενός μοντέλου, από ένα τομέα στον άλλο, μεταφέροντας πληροφορία από τον σχετικό τομέα. Ας θεωρηθεί το παρακάτω παράδειγμα, όπου 2 άτομα προσπαθούν να μάθουν πιάνο. Ο ένας δεν έχει καθόλου εμπειρία στη μουσική και ο άλλος έχει εκτενή εμπειρία στο να παίζει κιθάρα. Το άτομο με που έχει ήδη γνώση στη μουσική (κιθάρα) θα μπορέσει να μάθει πιάνο πιο αποτελεσματικά, χρησιμοποιώντας τη γνώση που έχει ήδη στην κιθάρα. Το

πιάνο και η κιθάρα είναι υποτομείς της Μουσικής. Αυτό το φαινόμενο ονομάζεται *Transfer Learning* όπου ο μαθητευόμενος χρησιμοποιεί τη γνώση που έχει σε ένα τομέα, για να μάθει πιο γρήγορα και αποτελεσματικά έναν άλλο σχετικό τομέα. Το παραπάνω παράδειγμα ισχύει και στην περίπτωση της βαθιάς μηχανικής μάθησης (Weiss et al., 2016), όπου ερευνητές χρησιμοποιούν ένα μοντέλο που εκπαιδεύτηκε ήδη σε ένα μεγάλο σετ δεδομένων, πάνω στο δικό τους σετ που ενδέχεται να είναι μικρότερο.

Ποια είναι η πρακτική χρησιμότητα του *Transfer Learning* και γιατί είναι χρήσιμο στο χώρο της μηχανικής μάθησης; Η εν λόγω μέθοδος χρησιμοποιείται όταν τα δεδομένα του *training set* δεν είναι επαρκής. Αν τα δεδομένα του *training set* δεν είναι αρκετά, το μοντέλο ενδέχεται να μην μπορέσει να μάθει τα απαραίτητα χαρακτηριστικά που χρειάζεται για την επίλυση του προβλήματος. Παρακάτω θα ακολουθήσει μία μικρή αναφορά στο τι μαθαίνει ένα *convolutional neural network*.



**Εικόνα 7-7: Παράδειγμα εκμάθησης ενός μοντέλου σε κάθε επίπεδο. Οι εικόνες πάρθηκαν από το βιβλίο *Deep Learning with python* (Chollet, 2018).**

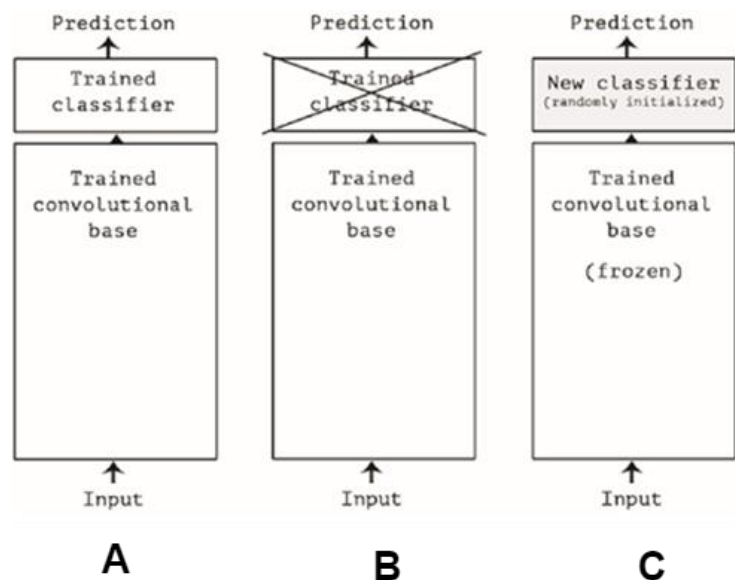
Κάθε επίπεδο (*layer*) ενός CNN μαθαίνει κάποιο χαρακτηριστικό της εικόνας. Για παράδειγμα, τα αρχικά επίπεδα μαθαίνουν οριζόντιες ή κάθετες ακμές, τα μεσαία επίπεδα μαθαίνουν σχήματα όπως κύκλους και ορθογώνια, τα τελευταία επίπεδα συνδυάζουν αυτή τη γνώση για την αναγνώριση και ταξινόμηση της εικόνας (βλ. Εικόνα 7-7).

Η εικόνα 7-7 είναι ένα παράδειγμα του τι μαθαίνει το κάθε επίπεδο ενός μοντέλου βαθιάς μάθησης. Στην εικόνα A φαίνεται πως το μοντέλο μαθαίνει να ξεχωρίζει κάθετες και διαγώνιες ακμές, όπου

διακρίνονται τα αυτιά, το περίγραμμα των ματιών και η μύτη. Στην εικόνα B φαίνεται πως μαθαίνει να ξεχωρίζει κύκλους, όπου διακρίνονται οι κόρες των ματιών. Τα μεταγενέστερα επίπεδα μαθαίνουν πιο πολύπλοκα χαρακτηριστικά. Στο τέλος, ο ταξινομητής (*classifier*) χρησιμοποιεί τη γνώση που έμαθε το μοντέλο και κατηγοριοποιεί την εικόνα ως γάτα (αν το μοντέλο έμαθε σωστά) ή ως κάτι άλλο.

Κατά συνέπεια, ένα *pre-trained CNN*, είναι ένα δίκτυο που εκπαιδεύτηκε ήδη σε ένα μεγάλο σετ δεδομένων, για την ταξινόμηση εικόνων σε μεγάλη κλίμακα. Αν το εν λόγω σετ είναι αρκετά μεγάλο και γενικό ως προς τις εικόνες που περιέχει, τότε τα χαρακτηριστικά που έμαθε το *pre-trained network*

ενδέχεται να μπορούν να χρησιμοποιηθούν για διάφορα προβλήματα που σχετίζονται με τη μηχανική όραση (Chollet, 2018).

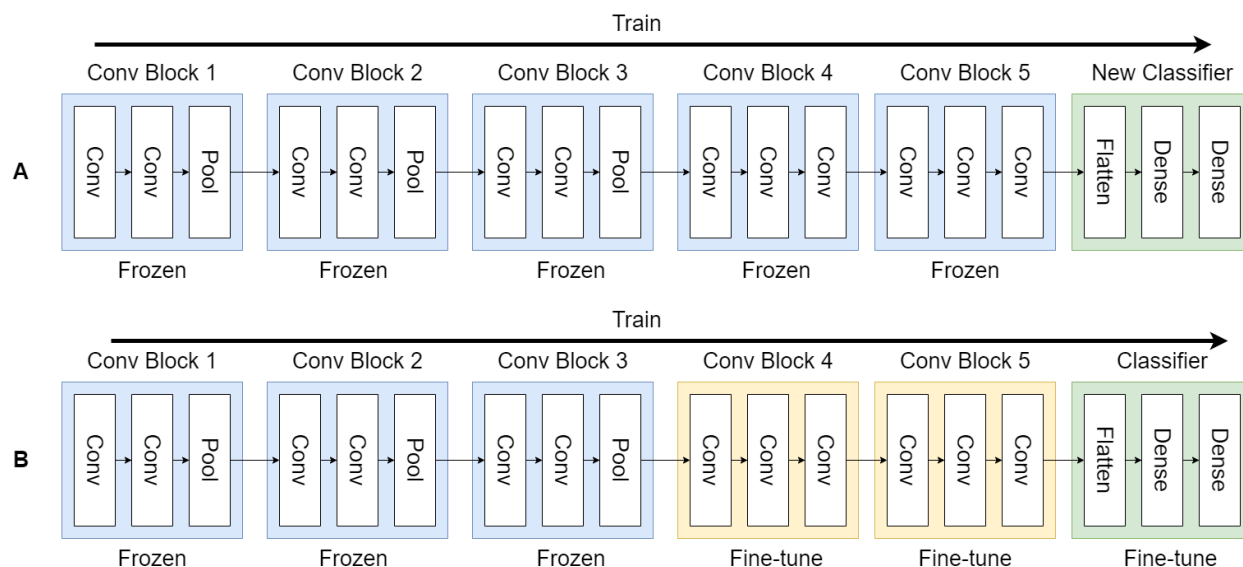


Τα αρχικά επίπεδα στα μοντέλα βαθιάς μάθησης μαθαίνουν χαρακτηριστικά χαμηλού επιπέδου (*low level features*) όπως αυτά που αναφέρθηκαν παραπάνω. Για αυτό τον λόγο, με τη χρήση *Transfer Learning* μπορεί ο ερευνητής, να χρησιμοποιήσει τα βάρη (*weights*) των αρχικών επιπέδων ενός άλλου μοντέλου, με αποτέλεσμα να μη χρειάζεται να αφιερώσει χρόνο και πόρους στην εκπαίδευση των εν λόγω επιπέδων.

Εικόνα 7-8: Συνοπτικά τα βήματα για την υλοποίηση TL (Chollet, 2018).

Όπως φαίνεται στην Εικόνα 7-8, στη φάση *A*, βρίσκεται το *pre-trained network*. Στη φάση *B*, αφαιρείται ο ταξινομητής (*classifier*) του αρχικού μοντέλου και τέλος, στη φάση *C* εισάγεται ο καινούριος ταξινομητής. Παράλληλα, όλα τα επίπεδα του μοντέλου εκτός το τελευταίο “παγώνουν”, περιορίζοντας την εκμάθησή τους. Επομένως, η εκπαίδευση γίνεται μόνο στο επίπεδο του ταξινομητή. Είναι καλή πρακτική να αλλάζει ο ταξινομητής και να μην χρησιμοποιείται ο αρχικός. Ο λόγος είναι πως ο αρχικός ενδέχεται να έχει εκπαιδευτεί σε άλλο σετ δεδομένων με πιθανότατα διαφορετικά *labels*. Για παράδειγμα, αν ο ταξινομητής ενός μοντέλου έχει εκπαιδευτεί στη ταξινόμηση εικόνων σκύλου και γάτας, τότε η γνώση που έχει δεν μπορεί να χρησιμοποιηθεί για κάτι άλλο πέρα από την εν λόγω ταξινόμηση. Για αυτό πρέπει να αφαιρείται ο αρχικός ταξινομητής και να εκπαιδεύεται ένας καινούριος. Γιατί όμως δεν ισχύει το ίδιο για τα υπόλοιπα επίπεδα; Όπως αναφέρθηκε παραπάνω, τα χαρακτηριστικά που μαθαίνουν τα *convolutional layers* είναι πιο γενικευμένα με αποτέλεσμα να μπορούν να χρησιμοποιηθούν για διάφορα προβλήματα. Είναι γεγονός πως όλες οι εικόνες έχουν κάποια κοινά χαρακτηριστικά όπως ακμές, σχήματα και διάφορα μοτίβα (*patterns*) που μπορεί να μάθει ένα μοντέλο. Εξάλλου, η διαδικασία του *TL* δεν τελειώνει σε αυτή την φάση. Αφού “παγώσουν” τα επίπεδα και τοποθετηθεί ο καινούριος ταξινομητής, τότε εκπαιδεύεται το μοντέλο, αλλά στην ουσία, μόνο το τελευταίο επίπεδο μαθαίνει, όλα τα υπόλοιπα είναι “παγωμένα” με αποτέλεσμα να μην αλλάζουν τα βάρη (*weights*) τους.

Αφού ολοκληρωθεί η εκπαίδευση, ακολουθεί η φάση της βελτίωσης (fine-tuning) όπου “ξεπαγώνουν” μερικά επίπεδα. Δηλαδή τα βάρη τους μπορούν να αλλάξουν και το μοντέλο ξανά εκπαιδεύεται.



**Εικόνα 7-9:** Υλοποίηση *fine-tuning* όπου μερικά επίπεδα του μοντέλου “ξεπαγώνουν” και εκπαιδεύονται. Στη φάση A, τα επίπεδα του μοντέλου είναι “παγωμένα” και εκπαιδεύεται μόνο ο καινούριος ταξινομητής. Στη φάση B, “ξεπαγώνουν” μερικά επίπεδα και το μοντέλο ξανά εκπαιδεύεται.

Συνοψίζοντας, υπάρχουν 5 βασικά βήματα, όσο αφορά την υλοποίηση του Transfer Learning. 1) Προσθήκη του ταξινομητή, 2) “Πάγωμα” των βασικών επιπέδων, 3) Εκπαίδευση του ταξινομητή εφόσον το υπόλοιπο μοντέλο είναι frozen, 4) Αφού ολοκληρωθεί η εκπαίδευση, unfreeze μερικά επίπεδα όπως φαίνεται στο παραπάνω διάγραμμα, 5) Εκπαίδευση ξανά.

### 7.5.2. Προσδιορισμός μοντέλου / Model Definition

Σχετικά με τον προσδιορισμό του κάθε μοντέλου, δημιουργήθηκε ένα υπόδειγμα (*template*) το οποίο χρησιμοποιήθηκε για όλες τις αρχιτεκτονικές και μοντέλα. Αρχικά, καθορίζεται το βασικό μοντέλο (*base\_model*) με τις ανάλογες μεθόδους του *Tensorflow* και *Keras*. Συγκεκριμένα για τα μοντέλα *ResNet50*, *ResNet101*, *DenseNet121*, *DenseNet169* και *InceptionV3* χρησιμοποιήθηκαν οι λειτουργίες (*ResNet50V2*, n.d.), (*ResNet101V2*, n.d.), (*DenseNet121*, n.d., p. 121), (*DenseNet169*, n.d.) και (*InceptionV3*, n.d., p. 3) αντίστοιχα. Σε όλες τις εν λόγω λειτουργίες, πάρθηκαν τα βάρη του μοντέλου που εκπαιδεύτηκε πάνω στο σετ δεδομένων *ImageNet*. Έπειτα, αφαιρέθηκε το τελευταίο επίπεδο (*classifier*), με την παράμετρο *include\_top = False*, εφόσον θα έμπαινε ο καινούριος ταξινομητής. Τέλος, καθορίστηκε το σχήμα της εισόδου (*input\_shape*) ανάλογα με το μέγεθος των εικόνων και τις απαιτήσεις του κάθε μοντέλου.

Μετά τον καθορισμό του μοντέλου, ακολούθησε το πάγωμα των επιπέδων του *base\_model*, πράγμα που είναι απαραίτητο για τη χρήση *Transfer Learning*. Στη συνέχεια, χρησιμοποιήθηκε η λειτουργία (*Tf.Keras.Input*, n.d.) όπου προετοιμάζει τα δεδομένα που θα εισαχθούν στο μοντέλο. Ακολούθως, εφαρμόζεται *Data Augmentation* και η τελευταία προ-επεξεργασία πάνω στα εν λόγω δεδομένα. Η προ-επεξεργασία που εφαρμόζεται μετά το *Data Augmentation*, ουσιαστικά μετατρέπει τις τιμές των εικονοστοιχείων από -1 σε 1 όσο αφορά τις αρχιτεκτονικές *ResNet* και *Inception* και για την αρχιτεκτονική *DenseNet* οι τιμές μετατρέπονται από 0 σε 1. Οι λειτουργίες που είναι υπεύθυνες για την εν λόγω μετατροπή των τιμών είναι (*Resnet\_v2.Preprocess\_input*, n.d.), (*Inception\_v3.Preprocess\_input*, n.d.) και (*Densenet.Preprocess\_input*, n.d.) όσο αφορά τις *ResNet*, *Inception* και *DenseNet* αντίστοιχα.

Μετά τον καθορισμό του βασικού μοντέλου και των λειτουργιών προ-επεξεργασίας, ακολουθεί η δημιουργία του καινούριου ταξινομητή (*classifier*). Συγκεκριμένα, ο ταξινομητής απαρτίζεται από 3 επίπεδα. Το πρώτο είναι *Global Average Pooling* (*Tf.Keras.Layers.GlobalAveragePooling2D*, n.d.) ή στην περίπτωση του *InceptionV3*, αντί *Global Average Pooling* χρησιμοποιήθηκε ένα *Flatten layer* (*Tf.Keras.Layers.Flatten*, n.d.). Έπειτα υπάρχει το *Dropout layer* (*Tf.Keras.Layers.Dropout*, n.d.) και τέλος ακολουθεί ένα *Dense layer* (*Tf.Keras.Layers.Dense*, n.d.) με 3 *Units* εφόσον υπάρχουν 3 κατηγορίες ταξινόμησης.

Το *Dropout* επίπεδο μηδενίζει τυχαία μερικές τιμές της εξόδου του προηγούμενου επιπέδου, και αυτό καθορίζεται από την τιμή του δείκτη. Για την προκειμένη πτυχιακή η τιμή είναι στο 0.2 (20%). Αυτό σημαίνει πως κάθε μονάδα έχει 20% πιθανότητα στο να μηδενιστεί. Η χρήση του *Dropout* είναι μία εξαιρετική μέθοδος για την αποφυγή υπερπροσαρμογής του μοντέλου. Επίσης, για λειτουργία ενεργοποίησης (*activation function*) χρησιμοποιήθηκε το *softmax* (βλ. Εξίσωση 1) όπου μετατρέπει ένα διάνυσμα αριθμών σε ένα διάνυσμα πιθανοτήτων. Τέλος, για *kernel initializer* χρησιμοποιήθηκε *HeNormal* (He et al., 2015a). Το παραπάνω υπόδειγμα χρησιμοποιήθηκε για όλα τα μοντέλα με τις απαραίτητες αλλαγές που απαιτούσε κάθε αρχιτεκτονική.

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$$

**Εξίσωση 1: : Ο Softmax τύπος που χρησιμοποιήθηκε.**

### 7.5.3. Μετρήσεις αξιολόγησης κατά την εκπαίδευση

Όσο αφορά τις μετρήσεις αξιολόγησης κατά την εκπαίδευση του κάθε μοντέλου, χρησιμοποιήθηκε *Categorical Accuracy* (*CategoricalAccuracy*, n.d.), *Precision* (*Precision*, n.d.) και *Recall* (*Recall*, n.d.) μαζί με *TP*, *TN*, *FP*, και *FN* (*FalseNegatives*, n.d.; *FalsePositives*, n.d.; *TrueNegatives*, n.d.; *TruePositives*, n.d.). Σχετικά με τη μέθοδο βελτιστοποίησης (*optimizer*), χρησιμοποιήθηκε η *Adam* (Kingma & Ba, 2017) με

αρχικό learning rate =  $4e^{-3}$ , beta\_1 = 0.9, beta\_2 = 0.999 και epsilon =  $1e^{-07}$ . Τέλος, η μέθοδος απώλειας (loss) που αξιοποιήθηκε είναι η Categorical Cross entropy (CategoricalCrossentropy, n.d.). Οι εξισώσεις για κάθε μέτρηση φαίνονται παρακάτω.

*Categorical Accuracy*: Υπολογίζει τη συχνότητα όπου οι προβλέψεις του μοντέλου αντιστοιχούν στις πραγματικές τιμές των δεδομένων.

$$\text{Εξίσωση 2: Categorical Cross Entropy} = -\log\left(\frac{e^{s_p}}{\sum_j e^{s_j}}\right)$$

$$\text{Εξίσωση 3: Precision} = \frac{TP}{TP+FP}$$

$$\text{Εξίσωση 4: Recall} = \frac{TP}{TP+FN}$$

Εφόσον η ταξινόμηση δεν είναι δυαδική (*binary classification*) αλλά *multi-class classification*, ο τρόπος που πρέπει να υπολογισθούν οι μετρήσεις *TP*, *TN*, *FP*, *FN* είναι λίγο διαφορετικός. Σε συντομία, για κάθε κλάση πρέπει να υπολογισθούν οι εν λόγω μετρήσεις ξεχωριστά. Για το εν λόγω πρόβλημα, πρώτα θα παρθεί η κλάση “COVID-19”.

**Πίνακας 7-1: Παράδειγμα για υπολογισμό μετρήσεων ενός *multi-class classification* προβλήματος. Οριζόντια είναι οι τιμές που πρόβλεψε το μοντέλο και κάθετα είναι οι πραγματικές τιμές.**

	COVID-19	Non-Covid	Normal
COVID-19	TP	FN	FN
Non-Covid	FP	TN	TN
Normal	FP	TN	TN

Ακολούθως, θα πραγματοποιηθεί η ίδια διαδικασία για όλες τις κλάσεις και με τις εξισώσεις 2 και 3, μπορεί να υπολογισθεί το *Precision* και *Recall*.

#### 7.5.4. Επανακλήσεις (*Callbacks*)

Παράλληλα, κατά τη διάρκεια της εκπαίδευσης χρησιμοποιήθηκαν τα Callbacks: Model Checkpoint, Early Stopping, Reduce Learning Rate on Plateau, Tensorboard και CSVLogger. Συμπληρωματικά, το Model Checkpoint (ModelCheckpoint, n.d.) είναι ρυθμισμένο στο να αποθηκεύει τα βάρη του μοντέλου κάθε φορά που υπάρχει πτώση του validation loss. Έπειτα, το Early Stopping (EarlyStopping, n.d.), είναι υπεύθυνο για 2 λειτουργίες. Η 1<sup>η</sup> είναι η διακοπή της εκπαίδευσης, εφόσον δεν υπάρξει βελτίωση στο validation loss για 8 συνεχόμενα epochs (patience = 8) και η 2<sup>η</sup> είναι η επαναφορά των καλύτερων βαρών του μοντέλου. Με λίγα λόγια, σταματάει την εκπαίδευση, και επαναφέρει το μοντέλο στην καλύτερη του

κατάσταση. Στη συνέχεια, το Reduce Learning on Plateau (ReduceLROnPlateau, n.d.) μειώνει το learning rate του μοντέλου εφόσον δεν υπάρξει βελτίωση στο validation loss για 3 συνεχόμενα epochs (patience = 3). Συγκεκριμένα ο συντελεστής μείωσης για όλα τα μοντέλα είναι 0.2, όπου το καινούριο learning rate καθορίζεται από την Εξίσωση 5.

$$\text{Εξίσωση 5: } \text{new learning rate} = \text{lr} * \text{factor}$$

Το επόμενο callback που χρησιμοποιήθηκε ονομάζεται Tensorboard (TensorBoard, n.d.) το οποίο είναι ένα εργαλείο για την οπτικοποίηση του μοντέλου κατά τη διάρκεια της εκπαίδευσης. Το εν λόγω εργαλείο προσφέρει ποικίλες λειτουργίες. Στην προκειμένη πτυχιακή θα χρησιμοποιηθεί για την παρακολούθηση της προόδου του μοντέλου κατά την εκπαίδευση του και για την επίβλεψη των μετρήσεων που αναφέρθηκαν παραπάνω.

Τέλος, χρησιμοποιήθηκε το callback CSVLogger (CSVLogger, n.d.) το οποίο μεταδίδει τα αποτελέσματα της εκπαίδευσης σε ένα csv αρχείο το οποίο αποθηκεύεται στο δίσκο με την πάροδο του κάθε epoch. Συγκεκριμένα, κατά την εκπαίδευση του μοντέλου, τα αποτελέσματα των μετρήσεων αποθηκεύονται στην ανάλογη μεταβλητή που ονομάζει ο χρήστης. Όπως αναφέρθηκε παραπάνω, στην προκειμένη πτυχιακή χρησιμοποιήθηκε το Jupyter Notebook μέσω του VS Code. Δύστυχος, στην περίπτωση που ο υπολογιστής ή το πρόγραμμα καταρρεύσει και το Jupyter Notebook κλείσει, η μεταβλητή αυτή χάνεται και ως υποπροϊόν χάνεται το ιστορικό της εκπαίδευσης με αποτέλεσμα να μην είναι δυνατή η συνέχεια της εκπαίδευσης από το σημείο που έμεινε πριν κλείσει το πρόγραμμα. Η χρήση του εν λόγω callback δίνει στον χρήστη ένα αντίγραφο της μεταβλητής με αποτέλεσμα να είναι δυνατή η ανάκληση του ιστορικού της εκπαίδευσης.

## 7.6. Αρχική εκπαίδευση μοντέλου

Εφόσον καθορίστηκαν τα παραπάνω *callbacks*, μπορεί να ξεκινήσει η πρώτη φάση της εκπαίδευσης του εκάστοτε μοντέλου, όπου όλα τα επίπεδα εκτός το τελευταίο είναι “παγωμένα” (βλ. Διάγραμμα 6 - φάση A). Σχετικά με τις παραμέτρους της εκπαίδευσης, καθορίζεται το *train set* και *validation set*, όπου το μοντέλο θα τρέξει πρώτα σε όλο το *train set* και στη συνέχεια θα αξιολογηθεί αυτόματα στο *validation set*. Τέλος, ο αριθμός των *epochs* που καθορίστηκε είναι 100. Βέβαια, κανένα μοντέλο δεν εκπαιδεύτηκε για 100 *epochs* εφόσον υπάρχει το *callback Early Stopping*. Το κάθε μοντέλο ενδέχεται να χρειαστεί διαφορετικό χρόνο και αριθμό *epochs* για να φτάσει στην καλύτερη απόδοση του, για αυτό τον λόγο επιλέχθηκε ένας μεγάλος αριθμός, έτσι ώστε και τα 5 μοντέλα έχουν αρκετό περιθώριο για την εκμάθησή τους. Τα αποτελέσματα, χρόνος εκπαίδευσης και *epochs* του κάθε μοντέλου θα αναλυθούν στο κεφάλαιο *Αποτελέσματα*.

### 7.6.1. Fine-tuning

Αφού ολοκληρωθεί η πρώτη φάση της εκπαίδευσης, ακολουθεί η φάση του *fine-tuning*. Συγκεκριμένα όπως αναφέρθηκε παραπάνω, στο σημείο αυτό “ξεπαγώνουν” μερικά επίπεδα του μοντέλου, τα οποία θα εκπαιδευτούν ξανά, με τη διαφορά πως αυτή τη φορά, θα έχουν τη δυνατότητα εκμάθησης.

Κάθε αρχιτεκτονική είναι διαφορετική ως προς τον αριθμό επιπέδων που απαρτίζεται αλλά και ως προς τον σχεδιασμό της. Για αυτό τον λόγο χρειάστηκε χρόνος και μελέτη για την εύρεση του κατάλληλου αριθμού επιπέδων που θα “ξεπαγώσουν” σε κάθε μοντέλο. Στο παρακάτω πίνακα αναγράφονται αναλυτικά οι παράμετροι του κάθε μοντέλου αφού “ξεπάγωσαν” τα ανάλογα επίπεδα.

**Πίνακας 7-2: Συνοπτικά ο αριθμός εκπαιδεύσιμων παραμέτρων του κάθε μοντέλου αφού “ξεπάγωσαν” τα ανάλογα μοντέλα.**

	ResNet50	ResNet101	DenseNet121	DenseNet169	InceptionV3
Total Params	23,564,800	42,632,707	7,040,579	12,647,875	22,023,971
Trainable Params:	14,970,880	25,040,899	5,527,299	11,059,843	17,588,163
Non-trainable params:	8,593,920	17,591,80	1,513,280	1,588,032	4,435,808

Έπειτα το κάθε μοντέλο ξανά εκπαιδεύεται με τα ίδια *callbacks*, *loss function* και *metrics*. Επίσης, όσο αφορά τη μέθοδο βελτιστοποίησης, όπως και πριν χρησιμοποιήθηκε η *Adam* αλλά με *learning rate*  $4e^{-4}$  αντί  $4e^{-3}$ . Στη συνέχεια, το κάθε μοντέλο εκπαιδεύεται για 10 με 15 *epochs*. Μετά την εκπαίδευση, και εφόσον η απόδοση του κάθε μοντέλου είναι ικανοποιητική, ακολουθεί η φάση της αξιολόγησης. Παρακάτω, στον πίνακα 3 φαίνονται συνοπτικά οι παράμετροι που χρησιμοποιήθηκαν και για τις 2 φάσεις (A – B). Τέλος, όλα τα μοντέλα εκπαιδεύτηκαν σε υπολογιστή με επεξεργαστή *Ryzen 5600X*, μνήμη *RAM 16GB*, κάρτα γραφικών *AMD RX Vega 64* και *Windows 10*.

**Πίνακας 7-3: Οι τιμές των παραμέτρων που χρησιμοποιήθηκαν για όλα τα μοντέλα.**

Batch Size	16
Image Size	224,224,3 → DenseNet, ResNet 256,256,3 → InceptionV3
Initial Learning Rate	$4e^{-3}$
Early Stopping	Patience = 8
Reduce Learning Rate	Patience = 3, factor = 0.2
Initial Training epochs	100
Fine-tuned Learning Rate	$4e^{-4}$



## 8. Αποτελέσματα

Στο παρόν κεφάλαιο θα γίνει ανάλυση των αποτελεσμάτων της συγκεκριμένης πτυχιακής εργασίας. Παράλληλα θα συζητηθούν οι μετρήσεις και η απόδοση των εν λόγω μοντέλων. Όπως αναφέρθηκε στη μεθοδολογία, έχουν εκπαιδευτεί 5 μοντέλα. Το κάθε μοντέλο θα εκπαιδευτεί στο *train set (training)*, κατά τη διάρκεια της εκπαίδευσης θα αξιολογηθεί στο *validation set (validation)* και τέλος εφόσον η απόδοση του είναι ικανοποιητική, θα αξιολογηθεί στο *test set (evaluation)*.

Η δομή του κεφαλαίου είναι 1) Πρώτα θα συζητηθεί ο αριθμός εκπαιδευόμενων παραμέτρων και τα *epochs* της εκπαίδευσης του κάθε μοντέλου, 2) Στη συνέχεια θα παρουσιασθούν γραφήματα σχετικά με τις μετρήσεις *ακρίβεια*, *απώλεια*, *Precision* και *Recall* όσο αφορά το *training* και *validation* του μοντέλου, 3) Έπειτα θα ακολουθήσει ο πίνακας της τελικής αξιολόγησης του μοντέλου μαζί με το *confusion matrix*.

Ο πίνακας τελικής αξιολόγησης και *confusion matrix* δημιουργήθηκαν με το (*Sklearn Metrics Classification Report*, n.d.) και (*Sklearn Metrics Confusion Matrix*, n.d.) αντίστοιχα μέσω της βιβλιοθήκης *scikit-learn* (*Scikit-Learn*, n.d.). Η στήλη *Support* του πίνακα αξιολόγησης δείχνει τον αριθμό εικόνων σε κάθε κλάση. Επίσης, όσο αφορά το γράφημα του *confusion matrix* χρησιμοποιήθηκε το (*Seaborn.Heatmap*, n.d.) για τη δημιουργία του. Κάθε μοντέλο θα αναλυθεί παρακάτω σε ξεχωριστό υποτομήμα.

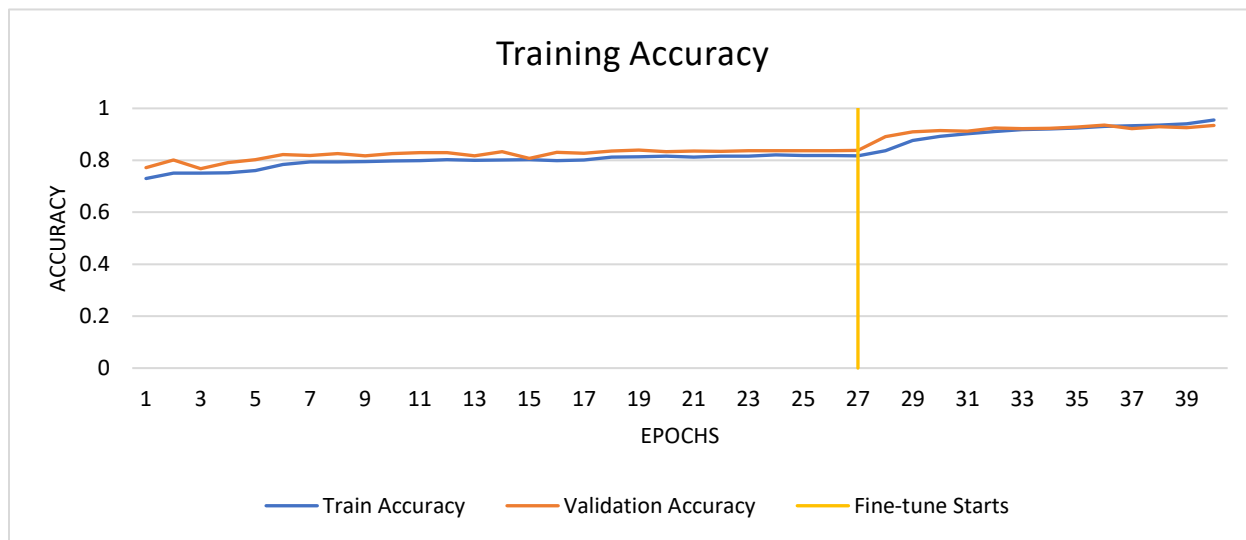
### 8.1. ResNet50

Το παρόν μοντέλο εκπαιδεύτηκε για συνολικά 40 *epochs*. Συγκεκριμένα, στη φάση **A** με “παγωμένα” *layers* εκπαιδεύτηκε για 27 *epochs*, και έπειτα στη φάση **B** εφαρμόστηκε *fine-tuning* για 13 *epochs*. Στο πίνακα παρακάτω φαίνεται αναλυτικά ο αριθμός εκπαιδευόμενων παραμέτρων πριν και μετά το *fine-tuning*.

*Πίνακας 8-1: Ο αριθμός εκπαιδευόμενων και μη παραμέτρων σε κάθε φάση του ResNet50.*

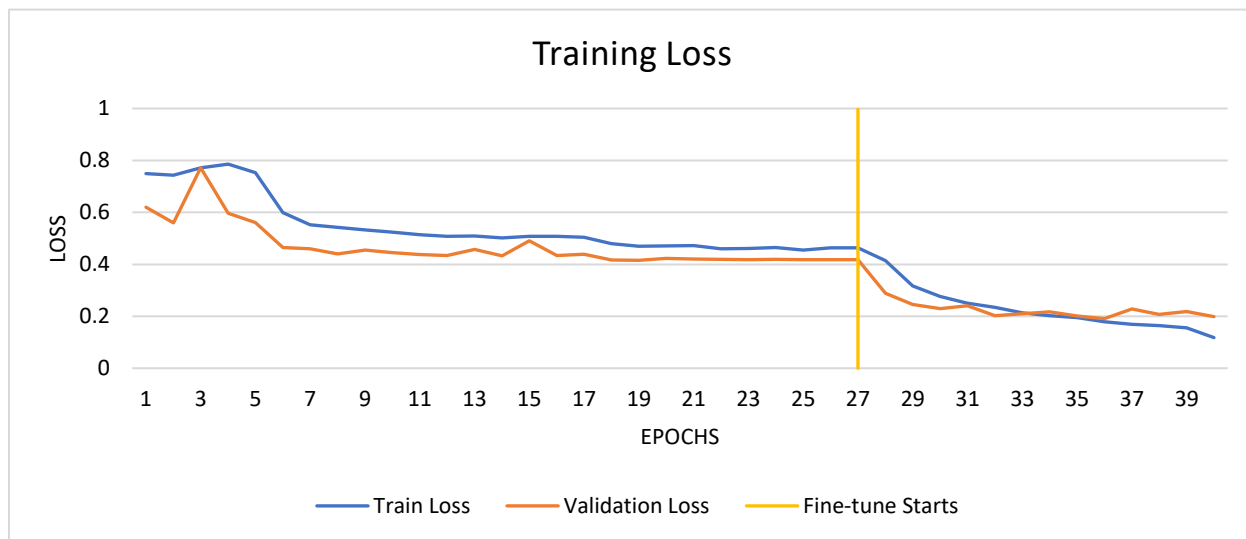
ResNet50	Trainable	Non-trainable	Total
Φάση <b>A</b>	6,147	23,564,800	
Φάση <b>B</b>	14,970,880	8,593,920	23,570,947

### 8.1.1. Αποτελέσματα εκπαίδευσης



Εικόνα 8-1: Η ακρίβεια του μοντέλου ResNet50.

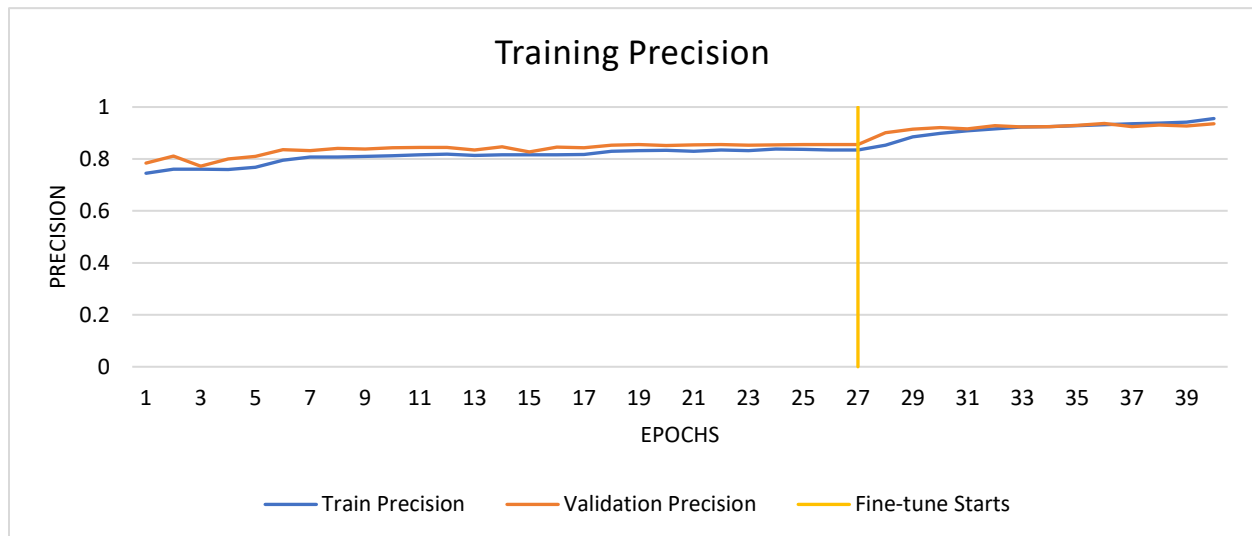
Στην παραπάνω εικόνα, φαίνεται η ακρίβεια του μοντέλου κατά την εκπαίδευση. Συγκεκριμένα στο *epoch* 27 - φάση A εξασφάλισε ακρίβεια **81.8%** (*train*) και **83.8%** (*validation*). Έπειτα, στη φάση B εφαρμόστηκε fine-tuning όπου φαίνεται μεγάλη άνοδος στην απόδοση του μοντέλου. Συγκεκριμένα, το μοντέλο συνεχίζει να βελτιώνεται μέχρι το *epoch* 40 όπου η ακρίβεια του μοντέλου φτάνει στο **95.5%** (*train*) και **93.4%** (*validation*).



Εικόνα 8-2: Η απώλεια του μοντέλου κατά την εκπαίδευση ResNet50.

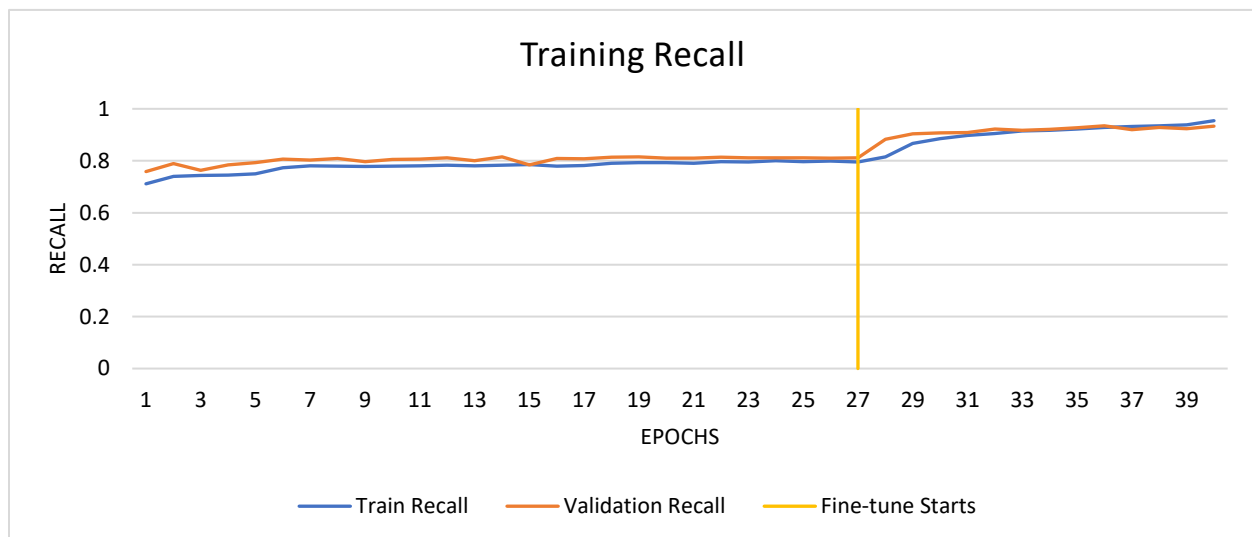
Στην παραπάνω εικόνα φαίνεται η απώλεια του μοντέλου κατά την εκπαίδευση. Συγκεκριμένα, στο *epoch* 27 – Φάση A, η απώλεια έπεσε στο **0.46** (*train*) και **0.42** (*validation*). Έπειτα, μετά την εφαρμογή *fine-tuning* παρατηρείται μεγάλη πτώση, όπου στο *epoch* 40 η απώλεια βρίσκεται στο **0.11** (*train*) και **0.19** (*validation*).

Στην παρακάτω εικόνα φαίνεται το *Precision* του μοντέλου. Συγκεκριμένα, στο *epoch* 27 – Φάση Α, το μοντέλο εξασφάλισε **83.5%** (*train*) και **85.6%** (*validation*). Παρομοίως, αφού εφαρμόσθηκε *fine-tuning*, στο *epoch* 40 το *precision* ανήλθε στο **95.6%** (*train*) και **93.6%** (*validation*).



*Εικόνα 8-3: Το Precision του μοντέλου ResNet50.*

Στην παρακάτω εικόνα φαίνεται το *Recall* του μοντέλου. Συγκεκριμένα, στο *epoch* 27 – Φάση Α, το μοντέλο εξασφάλισε **74.7%** (*train*) και **77.5%** (*validation*). Όπως και πριν, αφού εφαρμοσθεί *fine-tuning* στο *epoch* 30, παρατηρείται άνοδος, όπου το *Recall* ανήλθε στο **92.3%** (*train*) και **91.7%** (*validation*).



*Εικόνα 8-4: Το Recall του μοντέλου ResNet50.*

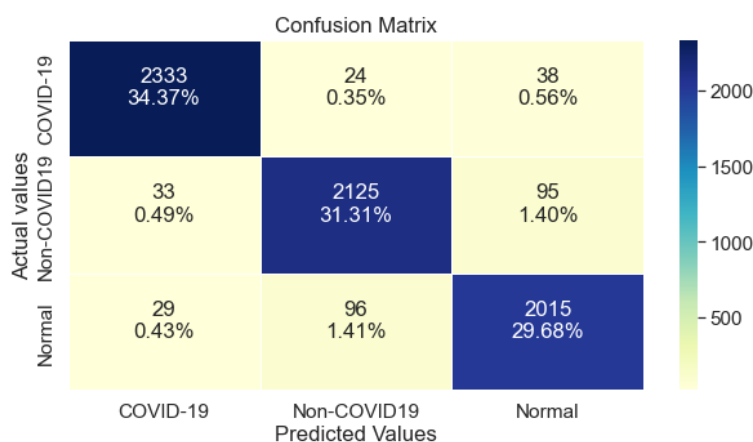
### 8.1.2. Αποτελέσματα της τελικής αξιολόγησης του *ResNet50*

*Πίνακας 8-2: Πίνακας τελικής αξιολόγησης του μοντέλου ResNet50.*

	Precision	Recall	F1-score	Support
COVID-19	0.97	0.97	0.97	2395
Non-COVID	0.95	0.94	0.94	2253
Normal	0.94	0.94	0.94	2140
Accuracy			0.95	6788
Macro avg	0.95	0.95	0.95	6788
Weighted avg	0.95	0.95	0.95	6788

Όπως φαίνεται παραπάνω στον *Πίνακα 8-2*, το μοντέλο κατά την τελική αξιολόγηση του, εξασφάλισε **97%** *Precision*, *Recall* και *f1-score* για την κλάση *COVID-19*. Επίσης, μπορεί να παρατηρηθεί πως η απόδοση του στις κλάσεις *Non-COVID* και *Normal* είναι λίγο μειωμένη όπου πέτυχε 94% στις εν λόγω μετρήσεις. Παράλληλα, το μοντέλο κατάφερε να εξασφαλίσει **95%** ακρίβεια, *macro average* και *weighted average*.

Στην *Εικόνα 8-5*, φαίνεται το *Confusion Matrix* του μοντέλου. Συγκεκριμένα, από τις 3 κλάσεις, παρατηρείται πως ο αριθμός λανθασμένων κατηγοριοποιήσεων του μοντέλου στις κλάσεις *Normal* και *Non-COVID* είναι ισότιμος. Τέλος, το μοντέλο απόδωσε καλύτερα στην κλάση *COVID-19*.



*Εικόνα 8-5: Το confusion matrix του μοντέλου ResNet50.*

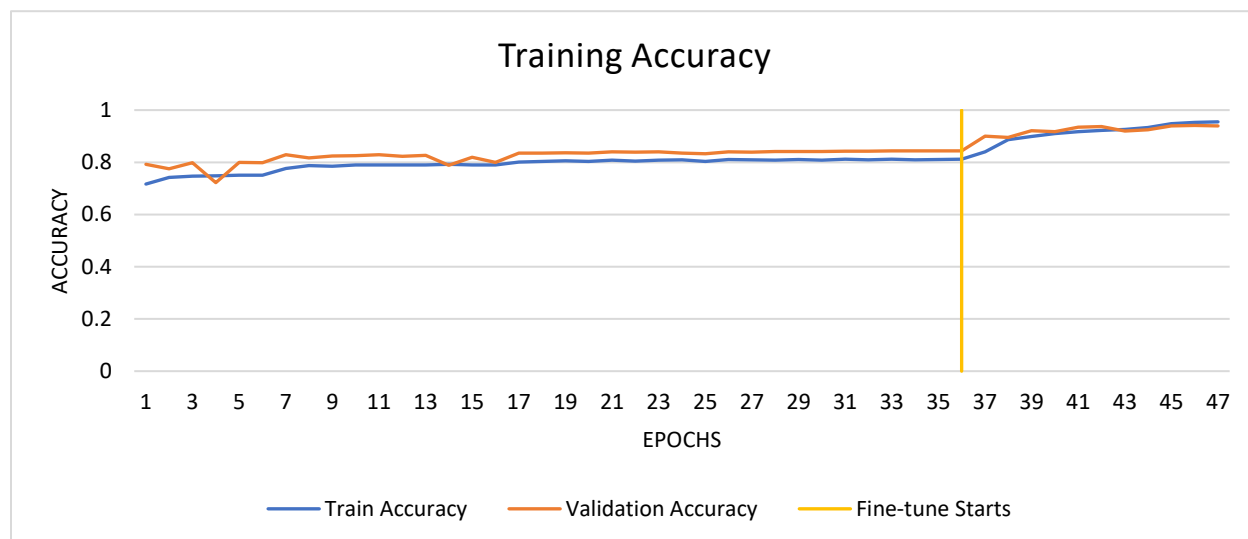
## 8.2. ResNet101

Το παρόν μοντέλο εκπαιδεύτηκε για συνολικά 47 *epochs*. Συγκεκριμένα, στη φάση **A** με “παγωμένα” *layers* εκπαιδεύτηκε για 36 *epochs*, και έπειτα στη φάση **B** εφαρμόστηκε *fine-tuning* για 11 *epochs*. Στο πίνακα παρακάτω φαίνεται αναλυτικά ο αριθμός εκπαιδευόμενων παραμέτρων πριν και μετά το *fine-tuning*.

Πίνακας 8-3: Ο αριθμός εκπαιδευόμενων και μη παραμέτρων σε κάθε φάση του ResNet101.

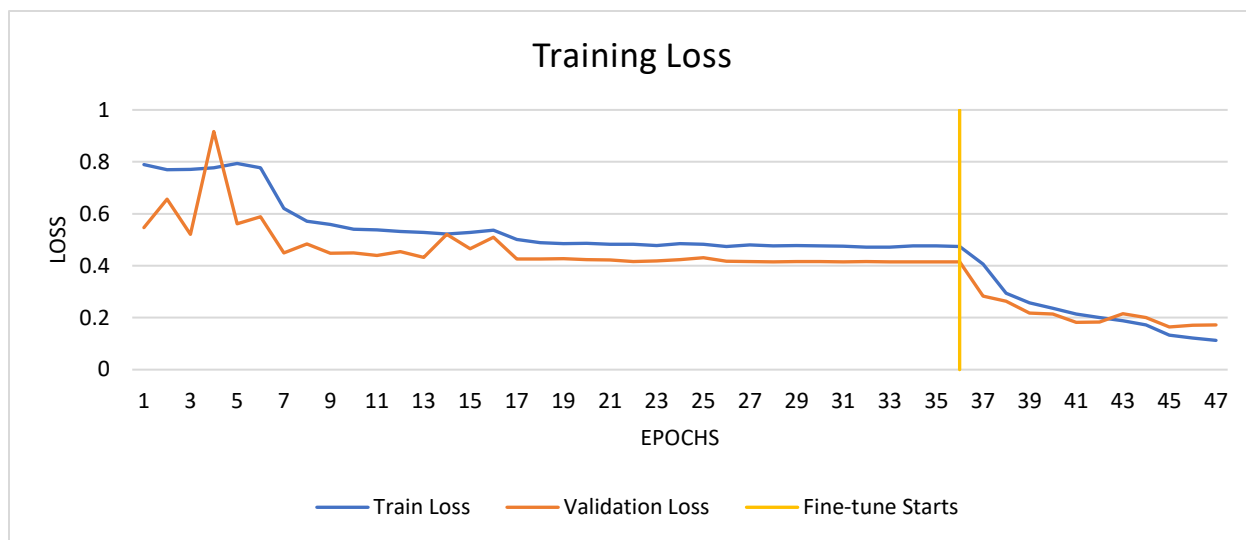
ResNet101	Trainable	Non-trainable	Total
Φάση <b>A</b>	6,147	42,626,560	
Φάση <b>B</b>	25,040,899	17,591,808	42,632,707

### 8.2.1. Αποτελέσματα εκπαίδευσης του ResNet101



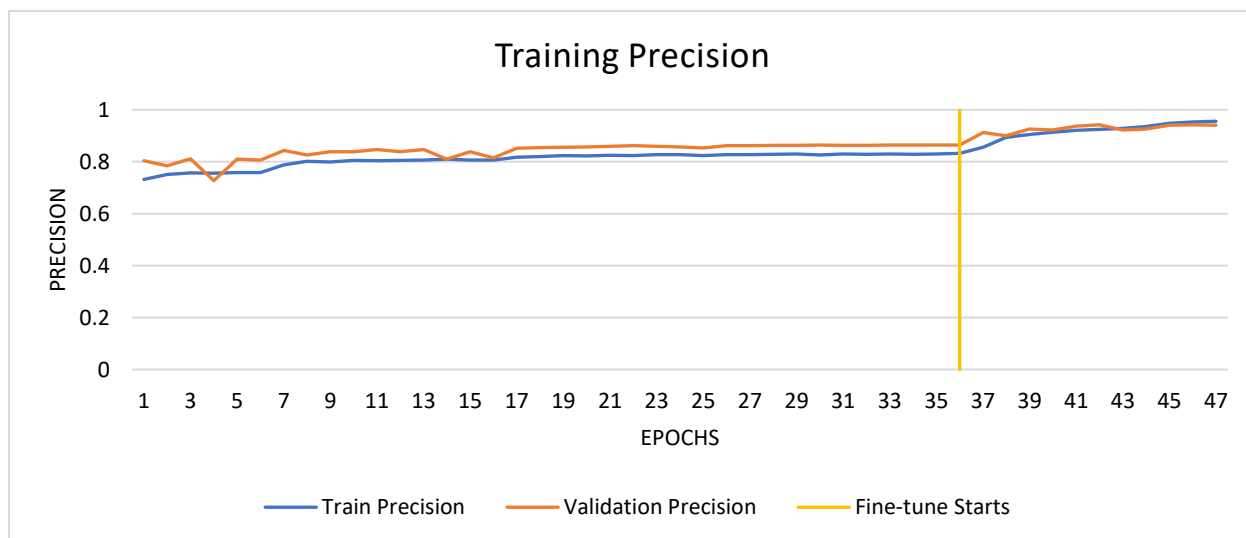
Εικόνα 8-6: Η ακρίβεια του μοντέλου ResNet101.

Στην παραπάνω εικόνα, φαίνεται η ακρίβεια του μοντέλου κατά την εκπαίδευση. Συγκεκριμένα στο *epoch* 27 - φάση **A** εξασφάλισε ακρίβεια **81.2%** (*train*) και **84.4%** (*validation*). Έπειτα, στη φάση **B** εφαρμόστηκε *fine-tuning* όπου φαίνεται μεγάλη άνοδος στην απόδοση του μοντέλου. Συγκεκριμένα, το μοντέλο συνεχίζει να βελτιώνεται μέχρι το *epoch* 47 όπου η ακρίβεια του μοντέλου φτάνει στο **95.5%** (*train*) και **94%** (*validation*).



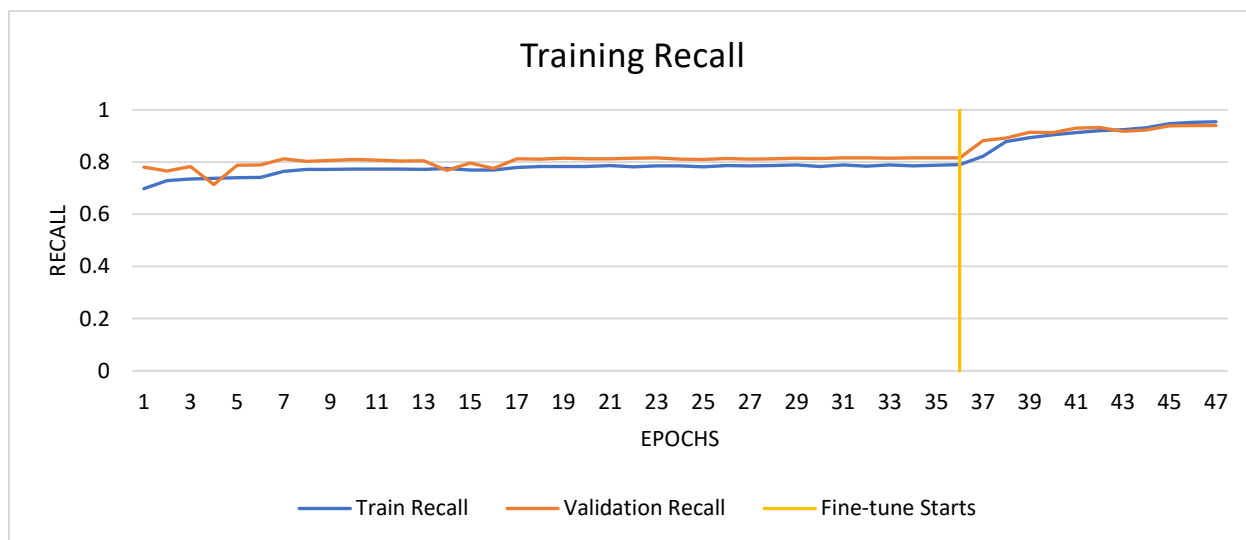
Εικόνα 8-7: Η απώλεια του μοντέλου ResNet101.

Στην εικόνα παραπάνω φαίνεται η απώλεια του μοντέλου κατά την εκπαίδευση. Συγκεκριμένα, στο *epoch* 27 – Φάση Α, η απώλεια έπεσε στο **0.47** (*train*) και **0.43** (*validation*). Έπειτα, μετά την εφαρμογή *fine-tuning* παρατηρείται μεγάλη πτώση, όπου στο *epoch* 47 η απώλεια βρίσκεται στο **0.11** (*train*) και **0.17** (*validation*).



Εικόνα 8-8: Το Precision του μοντέλου ResNet101.

Στην παραπάνω εικόνα φαίνεται το *Precision* του μοντέλου. Συγκεκριμένα, στο *epoch* 36 – Φάση Α, το μοντέλο εξασφάλισε **83.2%** (*train*) και **86.5%** (*validation*). Παρομοίως, αφού εφαρμόστηκε *fine-tuning*, στο *epoch* 47 το *precision* ανήλθε στο **95.6%** (*train*) και **94%** (*validation*).



Εικόνα 8-9: Το Recall του μοντέλου ResNet101.

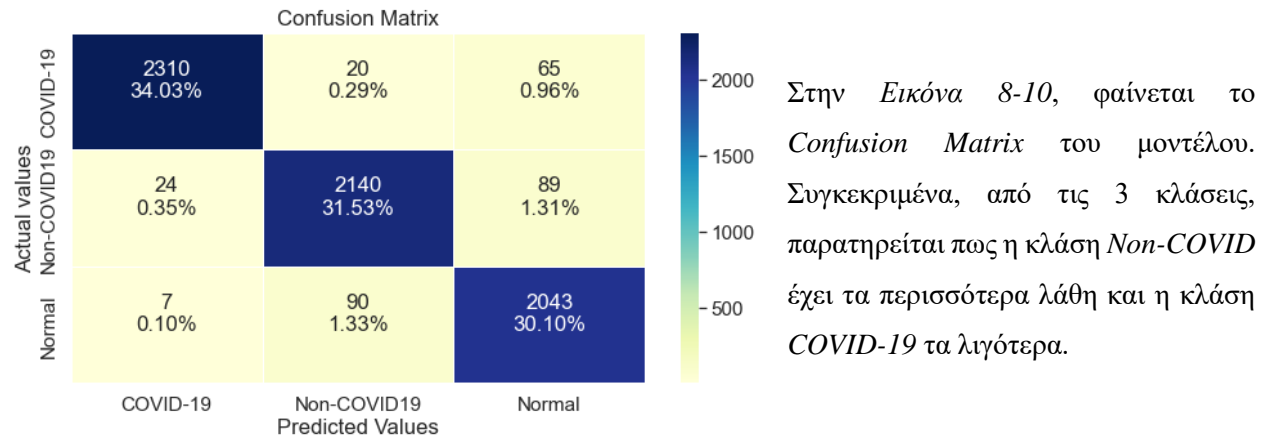
Στην παραπάνω εικόνα φαίνεται το *Recall* του μοντέλου. Συγκεκριμένα, στο *epoch* 36 – Φάση Α, το μοντέλο εξασφάλισε **69.8%** (*train*) και **78%** (*validation*). Όπως και πριν, αφού εφαρμοσθεί *fine-tuning* στο *epoch* 47, παρατηρείται άνοδος, όπου το *Recall* ανήλθε στο **95.5%** (*train*) και **93.9%** (*validation*).

#### 8.2.2. Αποτελέσματα της τελικής αξιολόγησης του ResNet101

Πίνακας 8-4: Πίνακας τελικής αξιολόγησης του μοντέλου ResNet101.

	Precision	Recall	F1-score	Support
COVID-19	0.99	0.96	0.98	2395
Non-COVID	0.95	0.95	0.95	2253
Normal	0.93	0.95	0.94	2140
Accuracy			0.96	6788
Macro avg	0.96	0.96	0.96	6788
Weighted avg	0.96	0.96	0.96	6788

Όπως φαίνεται παραπάνω στον Πίνακα 8-4, το μοντέλο κατά την τελική αξιολόγηση του, εξασφάλισε **99%** *Precision*, **96%** *Recall* και **98%** *f1-score* για την κλάση *COVID-19*. Επίσης μπορεί να παρατηρηθεί πως η απόδοση του στις κλάσεις *Non-COVID* και *Normal* είναι λίγο μειωμένη. Παράλληλα, το μοντέλο κατάφερε να εξασφαλίσει **96%** ακρίβεια, *macro average* και *weighted average*.



Εικόνα 8-10: Το confusion matrix του μοντέλου ResNet101.

### 8.3. DenseNet121

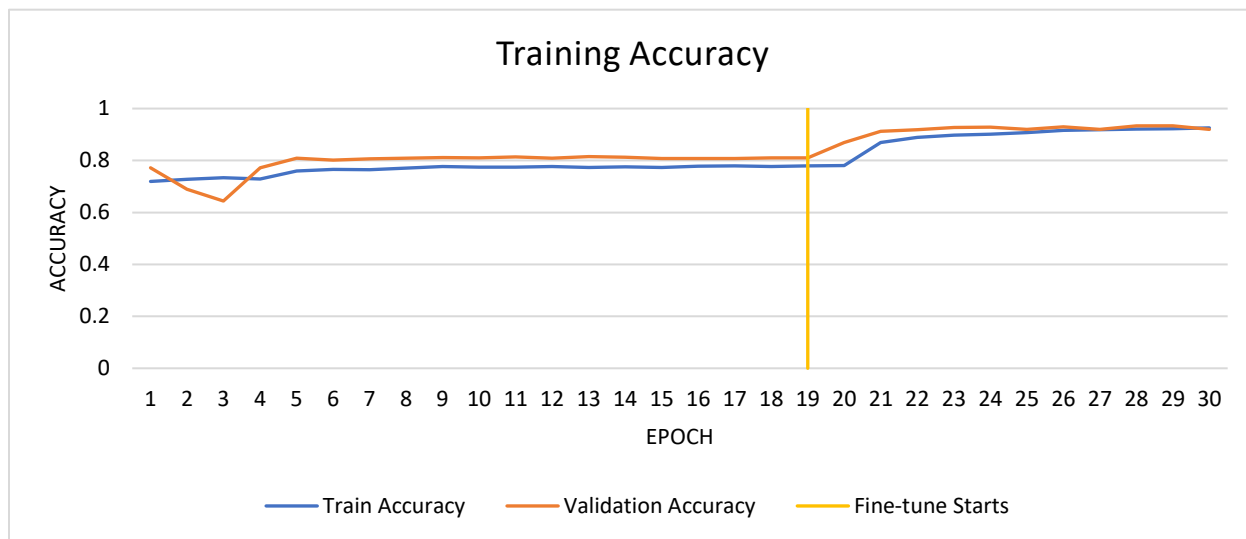
Το παρόν μοντέλο εκπαιδεύτηκε για συνολικά 30 *epochs*. Συγκεκριμένα, στη φάση **A** με “παγωμένα” *layers* εκπαιδεύτηκε για 19 *epochs*, και έπειτα στη φάση **B** εφαρμόστηκε *fine-tuning* για 11 *epochs*. Στο πίνακα παρακάτω φαίνεται αναλυτικά ο αριθμός εκπαιδευόμενων παραμέτρων πριν και μετά το *fine-tuning*.

Πίνακας 8-5: Ο αριθμός εκπαιδευόμενων και μη παραμέτρων σε κάθε φάση του DenseNet121.

DenseNet121	Trainable	Non-trainable	Total
Φάση <b>A</b>	3,075	7,037,504	
Φάση <b>B</b>	5,527,299	1,513,280	7,040,579

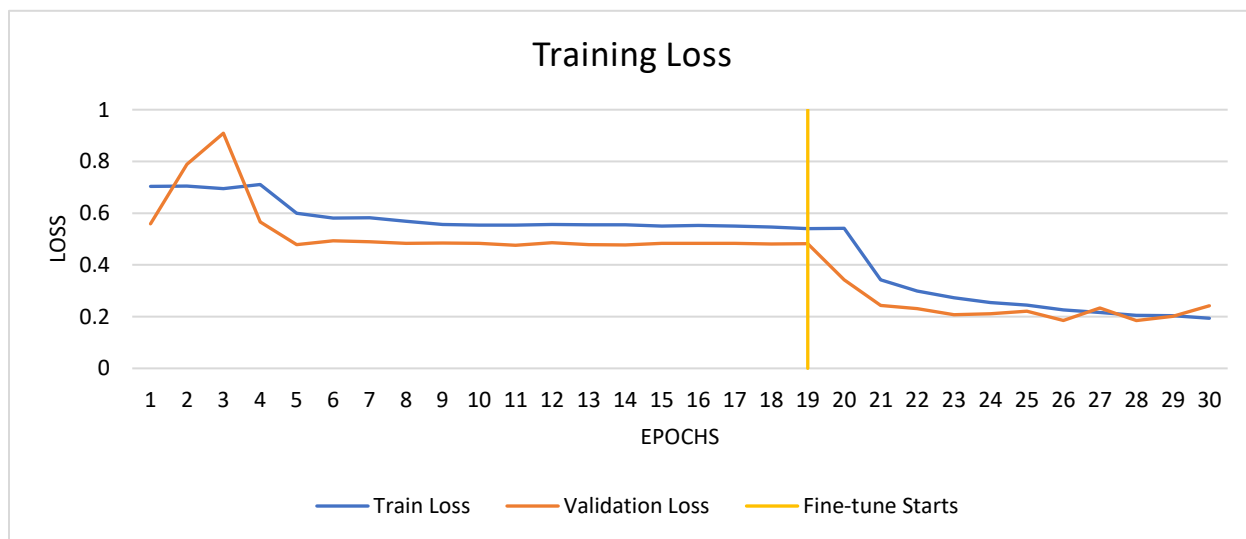


### 8.3.1. Αποτελέσματα εκπαίδευσης του *DenseNet121*



Εικόνα 8-11: Η ακρίβεια του μοντέλου *DenseNet121*.

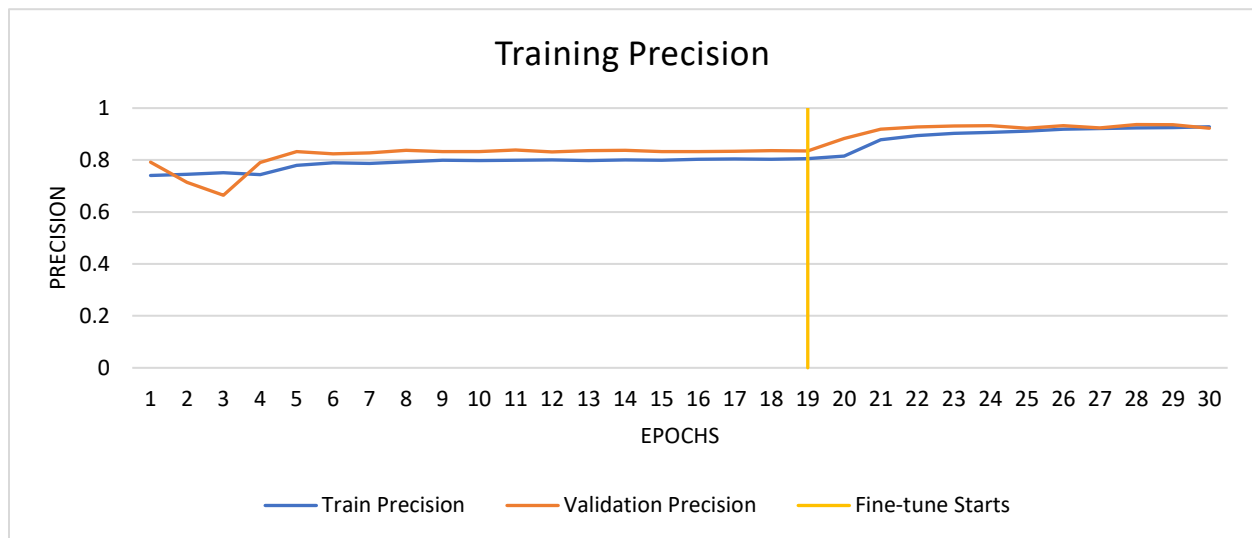
Στην παραπάνω εικόνα, φαίνεται η ακρίβεια του μοντέλου κατά την εκπαίδευση. Συγκεκριμένα στο *epoch* 19 - φάση Α εξασφάλισε ακρίβεια **78%** (*train*) και **81%** (*validation*). Έπειτα, στη φάση Β εφαρμόστηκε fine-tuning όπου φαίνεται μεγάλη άνοδος στην απόδοση του μοντέλου. Συγκεκριμένα, το μοντέλο συνεχίζει να βελτιώνεται μέχρι το *epoch* 30 όπου η ακρίβεια του μοντέλου φτάνει στο **92.3%** (*train*) και **92%** (*validation*).



Εικόνα 8-12: Η απώλεια του μοντέλου *DenseNet121*.

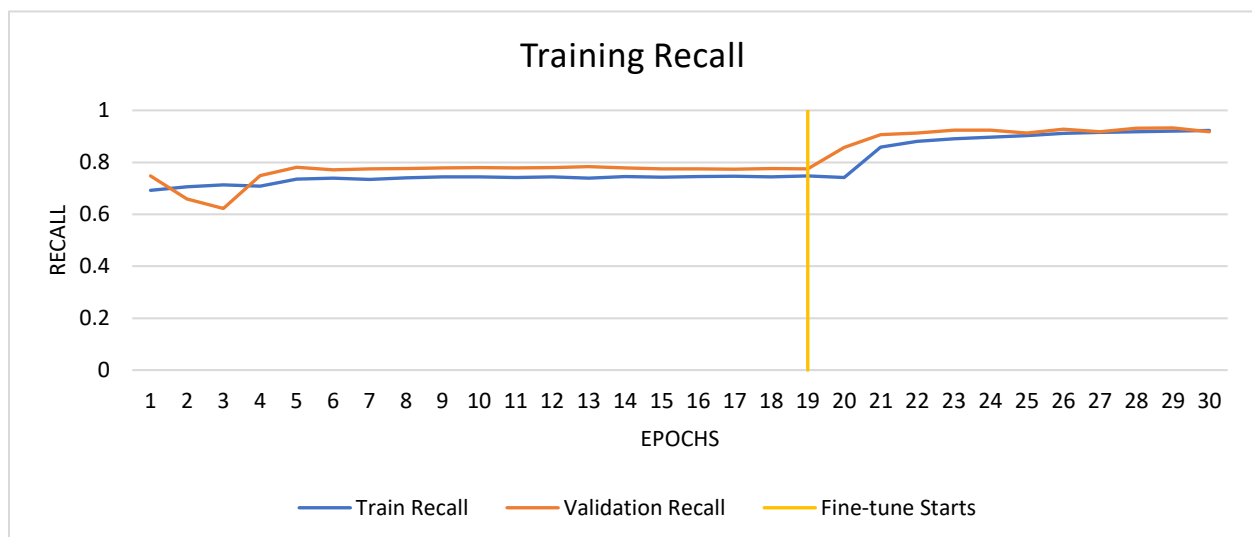
Στην εικόνα παραπάνω φαίνεται η απώλεια του μοντέλου κατά την εκπαίδευση. Συγκεκριμένα, στο *epoch* 19 – Φάση Α, η απώλεια έπεσε στο **0.54** (*train*) και **0.48** (*validation*). Έπειτα, μετά την εφαρμογή *fine-tuning* παρατηρείται μεγάλη πτώση, όπου στο *epoch* 30 η απώλεια βρίσκεται στο **0.19** (*train*) και **0.24** (*validation*).

Στην παρακάτω εικόνα φαίνεται το *Precision* του μοντέλου. Συγκεκριμένα, στο *epoch* 19 – Φάση Α, το μοντέλο εξασφάλισε **80%** (*train*) και **83%** (*validation*). Παρομοίως, αφού εφαρμόστηκε *fine-tuning*, στο *epoch* 30 το *precision* ανήλθε στο **92.8%** (*train*) και **92.2%** (*validation*).



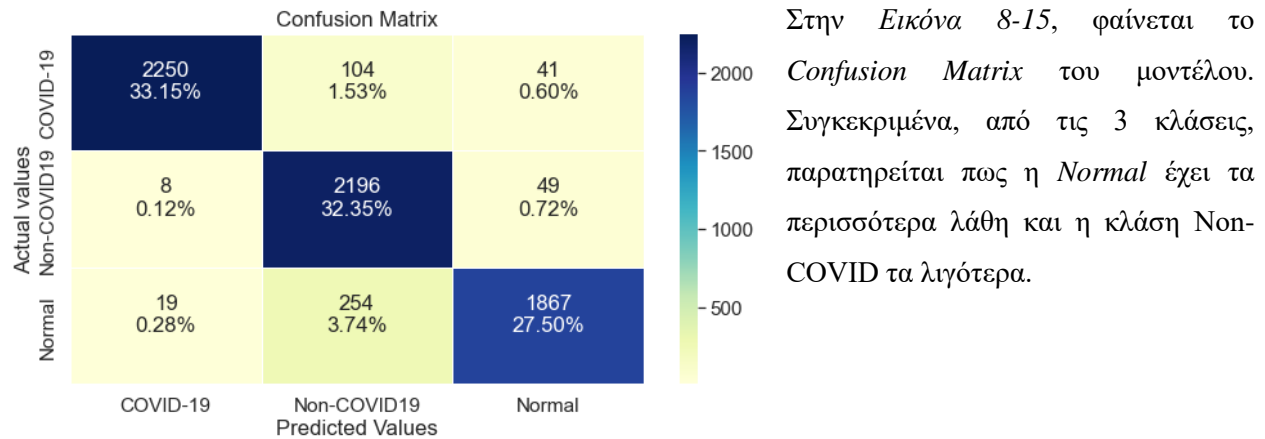
Εικόνα 8-13: Το *Precision* του μοντέλου *DenseNet121*.

Στην παρακάτω εικόνα φαίνεται το *Recall* του μοντέλου. Συγκεκριμένα, στο *epoch* 19 – Φάση Α, το μοντέλο εξασφάλισε **74.7%** (*train*) και **77.5%** (*validation*). Όπως και πριν, αφού εφαρμοσθεί *fine-tuning* στο *epoch* 30, παρατηρείται άνοδος, όπου το *Recall* ανήλθε στο **92.3%** (*train*) και **91.7%** (*validation*).



Εικόνα 8-14: Το *Recall* του μοντέλου *DenseNet121*.

### 8.3.2. Αποτελέσματα της τελικής αξιολόγησης του *DenseNet121*



Στην *Εικόνα 8-15*, φαίνεται το *Confusion Matrix* του μοντέλου. Συγκεκριμένα, από τις 3 κλάσεις, παρατηρείται πως η *Normal* έχει τα περισσότερα λάθη και η κλάση *Non-COVID* τα λιγότερα.

*Εικόνα 8-15: To confusion matrix του μοντέλου DenseNet121.*

*Πίνακας 8-6: Πίνακας τελικής αξιολόγησης του μοντέλου DenseNet121.*

	Precision	Recall	F1-score	Support
COVID-19	0.99	0.94	0.96	2395
Non-COVID	0.86	0.97	0.91	2253
Normal	0.95	0.87	0.91	2140
Accuracy			0.93	6788
Macro avg	0.93	0.93	0.93	6788
Weighted avg	0.93	0.93	0.93	6788

Όπως φαίνεται παραπάνω στον *Πίνακα 8-6*, το μοντέλο κατά την τελική αξιολόγηση του, εξασφάλισε **99% Precision**, **94% Recall** και **96% f1-score** για την κλάση *COVID-19*. Επίσης μπορεί να παρατηρηθεί το *Recall* στη κλάση *Normal* είναι μόνο **87%** και το *Precision* στη κλάση *Non-COVID* είναι μόνο **86%**. Παράλληλα, το μοντέλο κατάφερε να εξασφαλίσει **93%** ακρίβεια, *macro average* και *weighted average*.

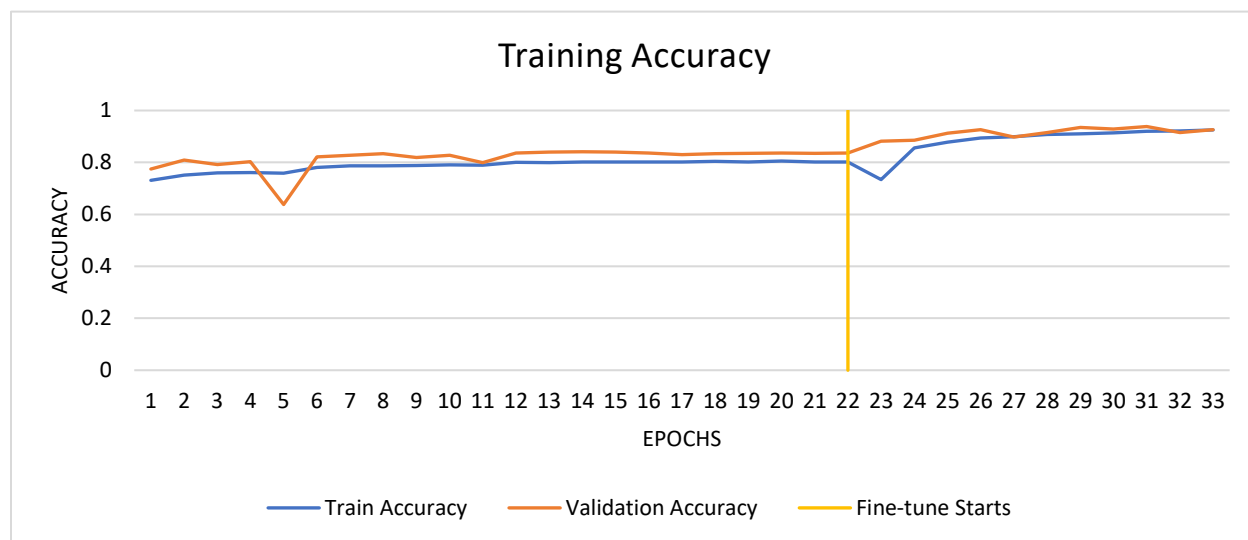
## 8.4. DenseNet 169

Το παρόν μοντέλο εκπαιδεύτηκε για συνολικά 33 *epochs*. Συγκεκριμένα, στη φάση **A** με “παγωμένα” *layers* εκπαιδεύτηκε για 22 *epochs*, και έπειτα στη φάση **B** εφαρμόστηκε *fine-tuning* για 11 *epochs*. Στο πίνακα παρακάτω φαίνεται αναλυτικά ο αριθμός εκπαιδευόμενων παραμέτρων πριν και μετά το *fine-tuning*.

*Πίνακας 8-7: Ο αριθμός εκπαιδευόμενων και μη παραμέτρων σε κάθε φάση του DenseNet169.*

DenseNet169	Trainable	Non-trainable	Total
Φάση <b>A</b>	4,995	12,642,880	
Φάση <b>B</b>	11,059,843	1,588,032	12,647,875

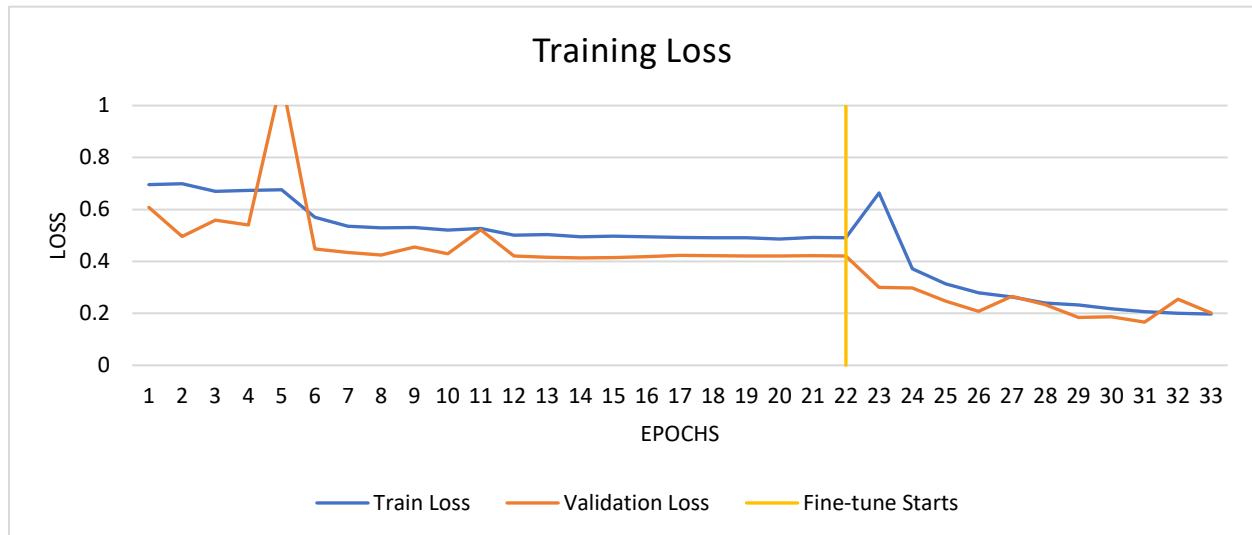
### 8.4.1. Αποτελέσματα εκπαίδευσης



*Εικόνα 8-16: Η ακρίβεια του μοντέλου DenseNet169.*

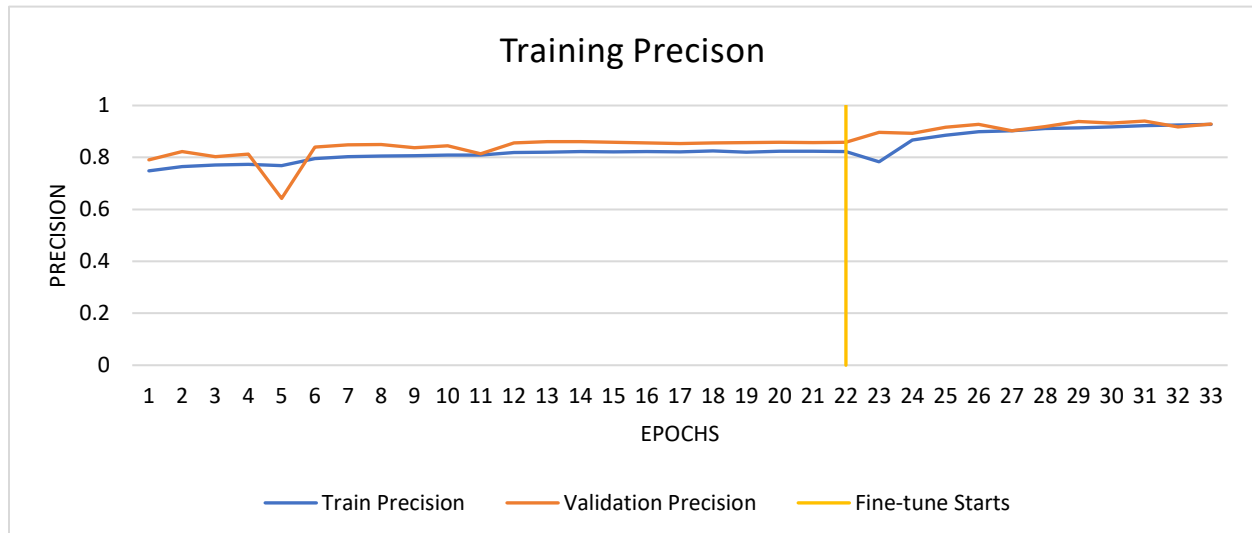
Στην εικόνα παραπάνω Διάγραμμα, φαίνεται η ακρίβεια του μοντέλου κατά την εκπαίδευση. Συγκεκριμένα στο *epoch* 22 - φάση **A** εξασφάλισε ακρίβεια **80%** (*train*) και **83.6%** (*validation*). Έπειτα, στη φάση **B** εφαρμόστηκε *fine-tuning* όπου φαίνεται μεγάλη άνοδος στην απόδοση του μοντέλου. Συγκεκριμένα, το μοντέλο συνεχίζει να βελτιώνεται μέχρι το *epoch* 33 όπου η ακρίβεια του μοντέλου φτάνει στο **92.5%** (*train*) και **92.6%** (*validation*).

Στην εικόνα παρακάτω φαίνεται η απώλεια του μοντέλου κατά την εκπαίδευση. Συγκεκριμένα, στο *epoch* 22 – Φάση Α, η απώλεια έπεσε στο **0.49** (*train*) και **0.42** (*validation*). Έπειτα, μετά την εφαρμογή *fine-tuning* παρατηρείται μεγάλη πτώση, όπου στο *epoch* 33 η απώλεια βρίσκεται στο **0.20** για *train* και *validation*.



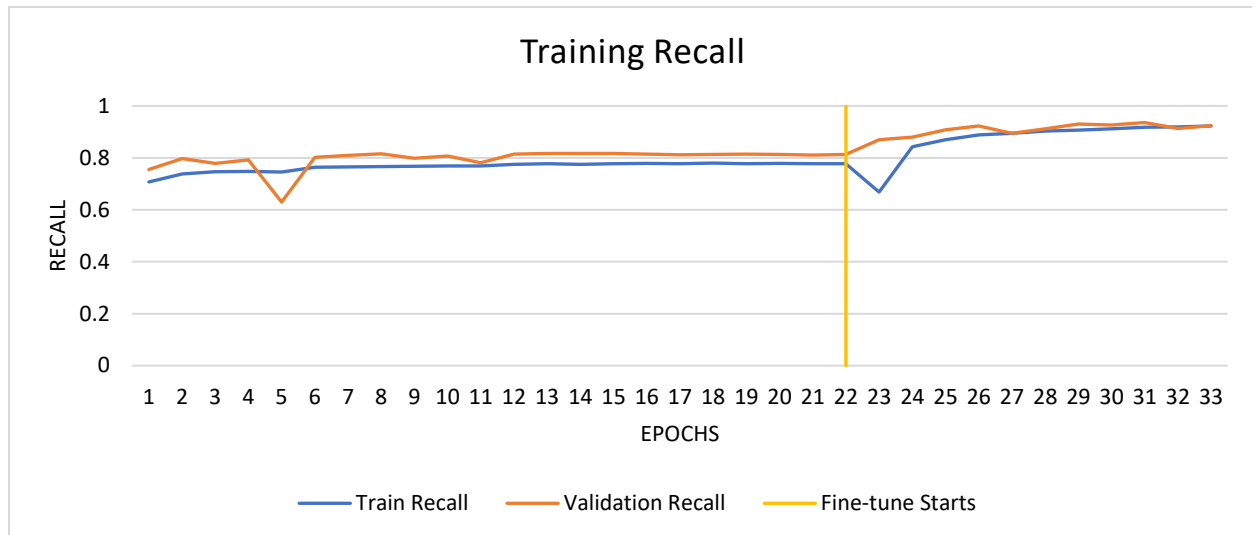
Εικόνα 8-17: Η απώλεια του μοντέλου DenseNet169.

Στην παρακάτω εικόνα φαίνεται το *Precision* του μοντέλου. Συγκεκριμένα, στο *epoch* 22 – Φάση Α, το μοντέλο εξασφάλισε **82%** (*train*) και **86%** (*validation*). Παρομοίως, αφού εφαρμόστηκε *fine-tuning*, στο *epoch* 33 το *precision* ανήλθε στο **92.7%** (*train*) και **92.9%** (*validation*).



Εικόνα 8-18: Το Precision του μοντέλου DenseNet169.

Στην παρακάτω εικόνα φαίνεται το *Recall* του μοντέλου. Συγκεκριμένα, στο *epoch* 22 – Φάση Α, το μοντέλο εξασφάλισε **77.8%** (*train*) και **81.4%** (*validation*). Όπως και πριν, αφού εφαρμοσθεί *fine-tuning* στο *epoch* 33, παρατηρείται άνοδος, όπου το *Recall* ανήλθε στο **92.3%** (*train*) και **92.4%** (*validation*).



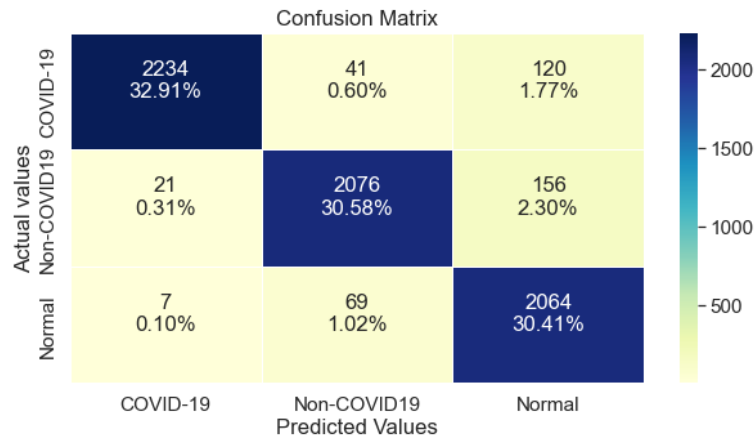
Εικόνα 8-19: Το *Recall* του μοντέλου *DenseNet169*.

#### 8.4.2. Αποτελέσματα της τελικής αξιολόγησης του *DenseNet169*

Πίνακας 8-8: Πίνακας τελικής αξιολόγησης του μοντέλου *DenseNet169*.

	Precision	Recall	F1-score	Support
COVID-19	0.99	0.93	0.96	2395
Non-COVID	0.95	0.92	0.94	2253
Normal	0.88	0.96	0.92	2140
Accuracy			0.94	6788
Macro avg	0.94	0.94	0.94	6788
Weighted avg	0.94	0.94	0.94	6788

Όπως φαίνεται παραπάνω στον Πίνακα 8-8, το μοντέλο κατά την τελική αξιολόγηση του, εξασφάλισε **99%** *Precision*, **93%** *Recall* και **96%** *f1-score* για την κλάση *COVID-19*. Επίσης, το μοντέλο φαίνεται να έχει μικρές αδυναμίες όσο αφορά την κλάση *Normal* όπου εξασφάλισε μόνο **88%** *Precision*. Παράλληλα, το μοντέλο κατάφερε να εξασφαλίσει **94%** ακρίβεια, *macro average* και *weighted average*.



Στην *Εικόνα 8-20*, φαίνεται το *Confusion Matrix* του μοντέλου. Συγκεκριμένα, από τις 3 κλάσεις, παρατηρείται πως η *Non-COVID* έχει τα περισσότερα λάθη και η κλάση *Normal* τα λιγότερα.

*Εικόνα 8-20: Το confusion matrix του μοντέλου DenseNet169.*

## 8.5. Inception-V3

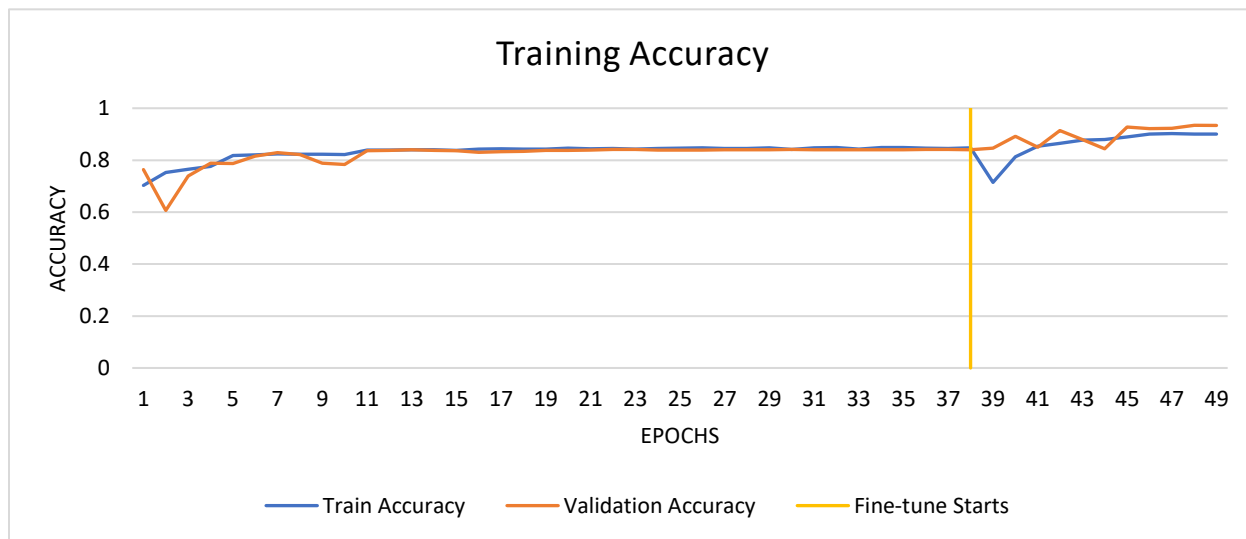
Το παρόν μοντέλο εκπαιδεύτηκε για συνολικά 49 *epochs*. Συγκεκριμένα, στη φάση **A** με “παγωμένα” *layers* εκπαιδεύτηκε για 38 *epochs*, και έπειτα στη φάση **B** εφαρμόστηκε *fine-tuning* για 11 *epochs*. Στο πίνακα παρακάτω φαίνεται αναλυτικά ο αριθμός εκπαιδευόμενων παραμέτρων πριν και μετά το *fine-tuning*.

*Πίνακας 8-9: Ο αριθμός εκπαιδευόμενων και μη παραμέτρων σε κάθε φάση του InceptionV3.*

InceptionV3	Trainable	Non-trainable	Total
Φάση <b>A</b>	221,187	21,802,784	
Φάση <b>B</b>	17,588,163	4,435,808	22,023,971

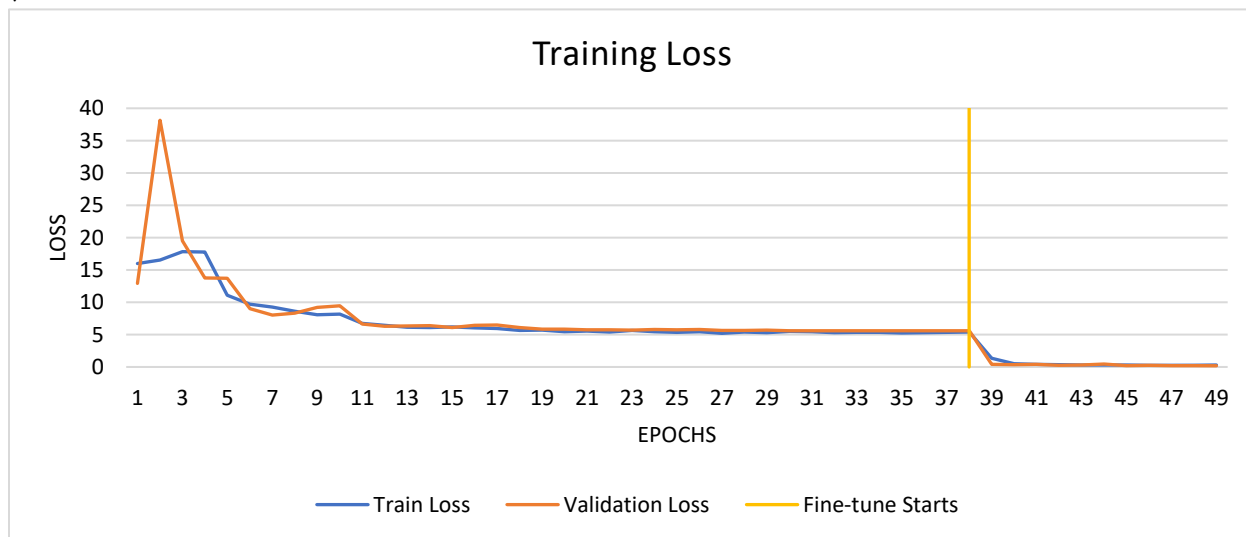
### 8.5.1. Αποτελέσματα εκπαίδευσης

Στην παρακάτω εικόνα, φαίνεται η ακρίβεια του μοντέλου κατά την εκπαίδευση. Συγκεκριμένα στο *epoch* 38 - φάση **A** εξασφάλισε ακρίβεια **84.7%** (*train*) και **84.1%** (*validation*). Έπειτα, στη φάση **B** εφαρμόστηκε fine-tuning όπου φαίνεται μεγάλη άνοδος στην απόδοση του μοντέλου. Συγκεκριμένα, το μοντέλο συνεχίζει να βελτιώνεται μέχρι το *epoch* 49 όπου η ακρίβεια του μοντέλου φτάνει στο **90%** (*train*) και **93.3%** (*validation*).



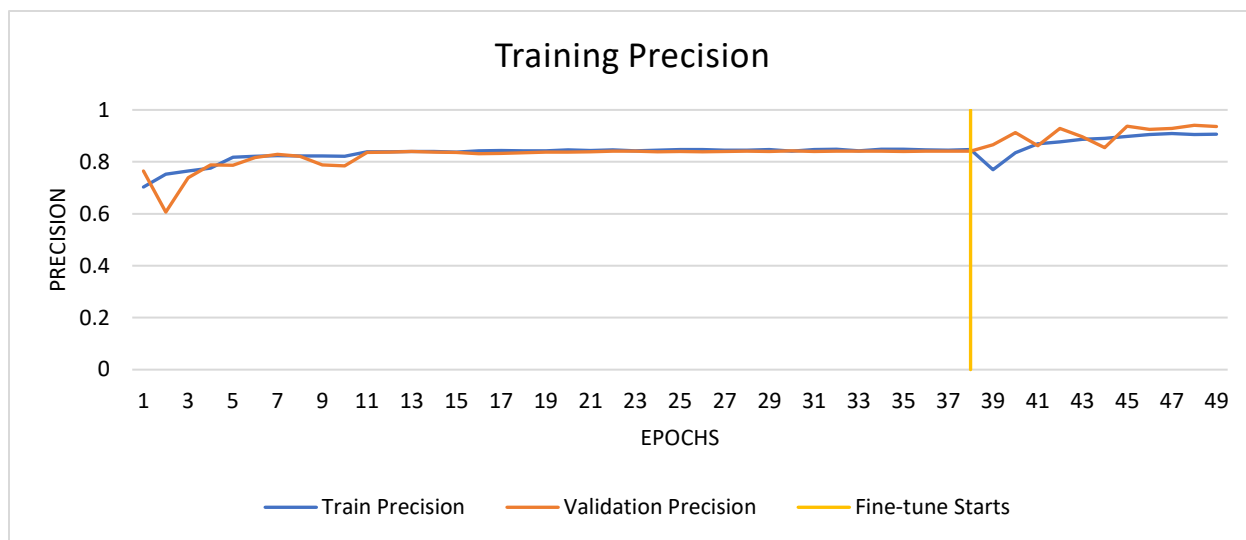
Εικόνα 8-21: Η ακρίβεια του μοντέλου InceptionV3.

Στην εικόνα παρακάτω φαίνεται η απώλεια του μοντέλου κατά την εκπαίδευση. Συγκεκριμένα, στο *epoch* 38 – Φάση **A**, η απώλεια έπεσε στο **5.42** (*train*) και **5.61** (*validation*). Έπειτα, μετά την εφαρμογή *fine-tuning* παρατηρείται μεγάλη πτώση, όπου στο *epoch* 49 η απώλεια βρίσκεται στο **0.28** για *train* και **0.18** για *validation*.



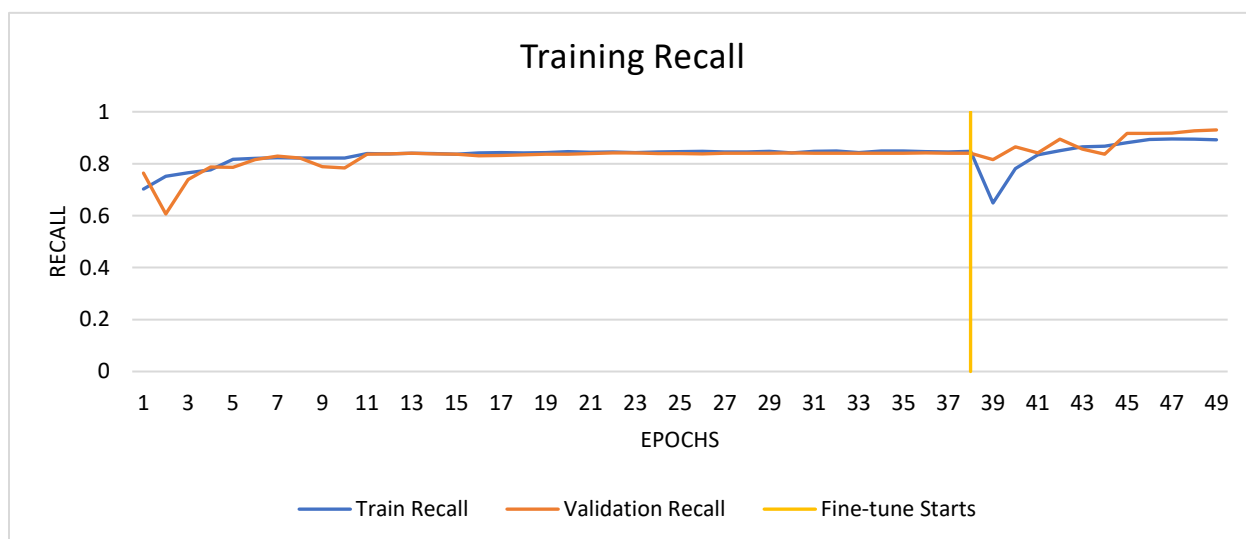
Εικόνα 8-22: Η απώλεια του μοντέλου InceptionV3.





Εικόνα 8-23: To Precision του μοντέλου InceptionV3.

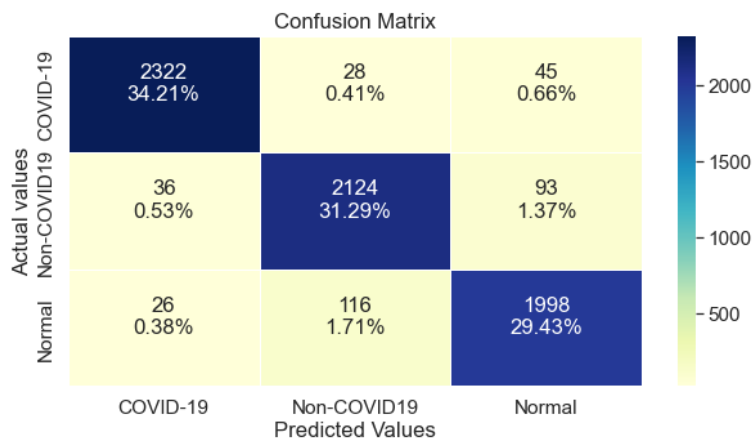
Στην παραπάνω εικόνα φαίνεται το *Precision* του μοντέλου. Συγκεκριμένα, στο *epoch* 38 – Φάση Α, το μοντέλο εξασφάλισε **84.7%** (*train*) και **84.1%** (*validation*). Παρομοίως, αφού εφαρμόσθηκε *fine-tuning*, στο *epoch* 49 το *precision* ανήλθε στο **90.6%** (*train*) και **93.6%** (*validation*).



Εικόνα 8-24: To Recall του μοντέλου InceptionV3.

Στην παραπάνω εικόνα φαίνεται το *Recall* του μοντέλου. Συγκεκριμένα, στο *epoch* 38 – Φάση Α, το μοντέλο εξασφάλισε **84.7%** (*train*) και **84.1%** (*validation*). Όπως και πριν, αφού εφαρμοσθεί *fine-tuning* στο *epoch* 49, παρατηρείται άνοδος, όπου το *Recall* ανήλθε στο **89.2%** (*train*) και **93%** (*validation*).

### 8.5.2. Αποτελέσματα της τελικής αξιολόγησης του *InceptionV3*



Στην *Εικόνα 8-25*, φαίνεται το *Confusion Matrix* του μοντέλου. Παρατηρείται πως η κλάση *COVID-19* έχει τα λιγότερα λάθη. Παράλληλα, οι κλάσεις *Normal* και *Non-COVID* είναι ισότιμες όσο αφορά τα λάθη που έγιναν από το μοντέλο.

*Εικόνα 8-25: Το confusion matrix του μοντέλου InceptionV3.*

*Πίνακας 8-10: Ο πίνακας τελικής αξιολόγησης του μοντέλου InceptionV3.*

	Precision	Recall	F1-score	Support
COVID-19	0.97	0.97	0.97	2395
Non-COVID	0.94	0.94	0.94	2253
Normal	0.94	0.93	0.93	2140
Accuracy			0.95	6788
Macro avg	0.95	0.95	0.95	6788
Weighted avg	0.95	0.95	0.95	6788

Όπως φαίνεται παραπάνω στον *Πίνακα 8-10*, το μοντέλο κατά την τελική αξιολόγηση του, εξασφάλισε **97% Precision**, **97% Recall** και **97% f1-score** για την κλάση *COVID-19*. Παράλληλα, παρατηρείται μικρότερη απόδοση στις κλάσεις *Non-COVID* και *Normal*. Τέλος, το μοντέλο κατάφερε να εξασφαλίσει **95%** ακρίβεια, *macro average* και *weighted average*.

## 9. Συζήτηση

Στο κεφάλαιο *Συζήτηση* θα γίνει αναφορά και σύγκριση των αποτελεσμάτων κάθε μοντέλου. Παράλληλα θα συζητηθούν οι περιορισμοί της Βαθιάς Μάθησης, τα προβλήματα που συναντήθηκαν στην εν λόγω πτυχιακή και οι λύσεις που εφαρμόστηκαν. Τέλος, θα αναφερθούν μελλοντικές αλλαγές που θα γίνουν στην προκειμένη πτυχιακή εργασία.

### 9.1. Αντιπαραβολή αποτελεσμάτων

Παρακάτω, στο *Πίνακα 9-1* φαίνονται συνοπτικά τα αποτελέσματα κάθε μοντέλου κατά την τελική τους αξιολόγηση. Στη στήλη “*Params*” φαίνεται ο αριθμός εκπαιδευόμενων παραμέτρων του κάθε μοντέλου μετά το fine-tuning. Το καλύτερο μοντέλο φαίνεται με **έντονη(\*)** γραφή.

*Πίνακας 9-1: Σύνοψη αποτελεσμάτων της τελικής αξιολόγησης του κάθε μοντέλου.*

Model	Ακρίβεια	Απώλεια	Precision	Recall	Params
ResNet50	95.4%	0.15	95.5%	95.3%	14,970,880
<b>ResNet101*</b>	<b>95.7%</b>	<b>0.13</b>	<b>95.7%</b>	<b>95.6%</b>	<b>25,040,899</b>
DenseNet121	93%	0.21	93.2%	92.8%	5,527,299
DenseNet169	93.9%	0.18	94.2%	93.8%	11,059,843
InceptionV3	94.9%	0.15	95.2%	94.6%	17,588,163

Τα αποτελέσματα του παραπάνω πίνακα πάρθηκαν με τη χρήση του (*Tf.Keras.Model.Evaluate*, n.d.). Όπως φαίνεται, το μοντέλο **ResNet101** έδειξε τα καλύτερα αποτελέσματα όπου εξασφάλισε **Ακρίβεια 95.7%**, **Απώλεια 0.13**, **Precision 95.7%** και **Recall 95.6%**. Βέβαια, αξίζει να σημειωθεί πως το εν λόγω μοντέλο είχε και τον μεγαλύτερο αριθμό εκπαιδευόμενων παραμέτρων, όπου αυτό συνεπάγεται σε μεγαλύτερη υπολογιστική χωρητικότητα (*computational capacity*).

Στο κεφάλαιο *Αποτελέσματα* φαίνονται οι μετρήσεις του κάθε μοντέλου κατά την εκπαίδευσή τους. Συγκεκριμένα, δεν παρατηρείται υπερπροσαρμογή (*overfitting*) από κανένα μοντέλο στο *training set*. Οι τιμές του κάθε μοντέλου ανάμεσα στο *train set* και *validation set* μεταβάλλονται αρμονικά και ισότιμα όπου οι διαφορές τους είναι μηδαμινή. Παράλληλα, μπορεί να παρατηρηθεί το φαινόμενο όπου και τα 5 μοντέλα, είχαν μειωμένη απόδοση όσο αφορά τις κλάσεις *Non-COVID* και *Normal* συγκριτικά με την κλάση *COVID-19*.

Αξιοσημείωτο είναι το γράφημα της απώλειας του μοντέλου *InceptionV3* (βλ. Διάγραμμα 8-19). Γενικότερα, το μοντέλο ξεκίνησε με τεράστια απώλεια όπου μετά από περίπου 5 *epochs* άρχισε να μειώνεται. Από το *epoch* 11 και μετά σταθεροποιήθηκε η πτώση της μέχρι το *epoch* 38. Κανένα άλλο από τα μοντέλα που εκπαιδεύτηκαν στην προκειμένη πτυχιακή εργασία ξεκίνησαν και διατήρησαν μία τόσο μεγάλη απώλεια μέχρι το σημείο του *fine-tuning*. Αδιαμφισβήτητα, η χρήση *TL* και *fine-tuning* βοήθησαν εξαιρετικά στην εκπαίδευση του μοντέλου. Αξίζει να σημειωθεί πως το εν λόγω μοντέλο παρουσίασε πολλά προβλήματα αφού έγινε η χρήση *fine-tuning*. Ο λόγος ήταν το *layer Global Average Pooling* στον ταξινομητή που χρησιμοποιήθηκε σε όλα τα μοντέλα. Μετά από αρκετά πειράματα, η λύση ήταν η αντικατάσταση του *Global Average Pooling layer* με ένα *Flatten layer* μόνο για το μοντέλο *InceptionV3*.

## 9.2. Σύγκριση αποτελεσμάτων

Στην έρευνα των (Narin et al., 2021) εκπαιδεύτηκαν 5 μοντέλα βαθιάς μάθησης για την αναγνώριση κορονοϊού. Τα εν λόγω μοντέλα είναι *InceptionV3*, *ResNet50*, *ResNet101*, *ResNet152* και *Inception-ResNetV2*. Το μοντέλο με την καλύτερη απόδοση ήταν το *ResNet50* όπου η ακρίβεια του έφτασε το **99.7%**, *Recall* **98.8%**, *Specificity* **99.8%**, *Precision* **98.3%** και *f1-score* **98.5%**.

Παράλληλα, οι (Asif et al., 2020) χρησιμοποίησαν ένα *pre-trained InceptionV3* μοντέλο για την αναγνώριση κορονοϊού. Η ακρίβεια του εν λόγω μοντέλου όσο αφορά την ταξινόμηση εικόνων κορονοϊού κατάφερε να φτάσει πάνω από **98%**.

Στη συνέχεια, οι (Keidar et al., 2021) εκπαιδεύσαν και αξιολόγησαν ξεχωριστά τα μοντέλα *Resnet34*, *ResNet50*, *ResNet152*, *CheXpert* και *VGG16* όπου η ακρίβεια του κάθε μοντέλου κυμαίνεται από **80.5%** μέχρι **90%**. Έπειτα, συνδυάζοντας τον μέσο όρο των αποτελεσμάτων από τα παραπάνω μοντέλα (*ensemble model*), έδειξαν ακρίβεια **90.3%**, *Sensitivity* **90.5%**, και *Specificity* **90.0%**.

Τέλος, οι (Arias-Garzón et al., 2021) χρησιμοποίησαν το μοντέλο *U-net* για *lung segmentation* και τα μοντέλα *VGG16* και *VGG19* όσο αφορά την αναγνώριση κορονοϊού. Συνοπτικά, η ακρίβεια που έδειξαν ήταν στο **92.72%** για την ταξινόμηση εικόνων σε *COVID* και *No-COVID*. Παράλληλα, όσο αφορά μόνο την ταξινόμηση της κλάσης *COVID*, η ακρίβεια ανήλθε στο **95.63%**.

Αξίζει να σημειωθεί πως τα παραπάνω αποτελέσματα δεν είναι καθοριστικά για όλα τα σετ δεδομένων. Για παράδειγμα, τα περισσότερα σετ δεδομένων μέχρι τώρα προέρχονται από Ευρωπαϊκές χώρες. Ασθενείς από τον υπόλοιπο κόσμο ενδέχεται να έχουν διαφορετικά χαρακτηριστικά στις ακτινογραφίες τους όσο αφορά τον τρόπο που πάρθηκαν οι εικόνες και τα μηχανήματα που χρησιμοποίησαν. Παράλληλα, εξαιτίας

των μεταλλάξεων του κορονοϊού, ενδέχεται τα παθολογικά χαρακτηριστικά να είναι διαφορετικά σε κάθε σετ δεδομένων.

### 9.3. Περιορισμοί της βαθιάς μηχανικής μάθησης

Παρόλο που η βαθιά μηχανική μάθηση έχει την δυνατότητα επίλυσης σύνθετων προβλημάτων, υπάρχουν μερικοί περιορισμοί στη χρήση της. Ο πρώτος περιορισμός είναι η υπολογιστική δύναμη που απαιτείται για την χρήση ενός μοντέλου βαθιάς μάθησης. Για παράδειγμα, στην εν λόγω πτυχιακή εργασία, το μοντέλο *ResNet101* χρειάστηκε περίπου 10 ώρες εκπαίδευσης συμπεριλαμβανομένου το *fine-tuning*. Βέβαια, αυτό δεν σημαίνει πως το μοντέλο εκπαιδεύτηκε μόνο 1 φορά. Συνολικά, για την εύρεση των καλύτερων παραμέτρων όπως *batch-size*, *learning rate*, *learning rate decay*, κ.λπ. έπρεπε να εκπαιδευτεί 7 φορές. Συνεπώς, μόνο για το *ResNet101* χρειάστηκαν περίπου 70 ώρες εκπαίδευσης. Αξίζει να σημειωθεί πως το συγκεκριμένο μοντέλο είχε μερικά *layers* “παγωμένα” πράγμα που έκανε την εκπαίδευση γρηγορότερη. Το σετ δεδομένων που χρησιμοποιήθηκε περιέχει μόνο ~30,000 εικόνες. Γενικότερα, υπάρχουν σετ δεδομένων με εκατομμύρια εικόνες όπου ο χρόνος εκπαίδευσης των μοντέλων είναι μήνες. Το εν λόγω πρόβλημα συνεπάγεται με το πρόβλημα της ποσότητας δεδομένων που χρειάζεται ένα μοντέλο για να μάθει. Είναι γνωστό πως τα βαθιά μοντέλα μάθησης χρειάζονται πολλά δεδομένα για να μάθουν. Αν το σετ δεδομένων δεν είναι επαρκές, τότε το μοντέλο ενδέχεται να μην μάθει ή να μάθει το σετ τόσο καλά όπου εμφανίζεται το πρόβλημα της υπερπροσαρμογής. Παράλληλα, άλλος περιορισμός των βαθιών μοντέλων μάθησης είναι το φαινόμενο του “μαύρου κουτιού”. Εξαιτίας των εκατομμυρίων παραμέτρων που απαρτίζεται το κάθε μοντέλο, είναι σχεδόν αδύνατον να ξέρει ο ερευνητής ακριβώς το τι μαθαίνει η κάθε παράμετρος.

### 9.4. Προβλήματα στην εν λόγω πτυχιακή εργασία

Όπως αναφέρθηκε παραπάνω, το μέγεθος του σετ δεδομένων είναι σημαντικό για την εκμάθηση ενός μοντέλου. Για την εν λόγω πτυχιακή εργασία, λήφθηκαν υπόψιν διάφορα σετ δεδομένων πριν επιλεγεί το σετ που χρησιμοποιήθηκε τώρα. Το μεγαλύτερο εμπόδιο σε κάθε σετ ήταν το μέγεθος του. Ο κορονοϊός είναι σχετικά μία πρόσφατη ασθένεια και οι εικόνες που υπήρχαν διαθέσιμες για ελεύθερη χρήση πριν 1 ή 2 χρόνια ήταν ανεπαρκής. Αυτό ήταν το πρώτο κριτήριο, το μέγεθος του σετ δεδομένων. Το δεύτερο κριτήριο ήταν η συμπερίληψη ενός υποσυνόλου του σετ (*test set*) για την τελική αξιολόγηση του μοντέλου. Όπως παρατηρείται στο κεφάλαιο *Αποτελέσματα* όλα τα μοντέλα κατά την εκπαίδευση τους, εξασφάλισαν  $92\% >$  σε όλες τις μετρήσεις. Αυτό όμως δεν ισχύει όταν αξιολογούνται σε δεδομένα που δεν έχουν ξανά δει. Για αυτό ήταν εξαιρετικά σημαντική η συμπερίληψη ενός *test set*. Βέβαια, η έλλειψη αξονικών

ακτινογραφιών θώρακα για κορονοϊό έκανε την εύρεση ενός σετ δεδομένων που να πληροί αυτά τα 2 κριτήρια εξαιρετικά δύσκολη. Το τελευταίο βασικό κριτήριο ήταν η κατανομή του σετ δεδομένων. Συγκεκριμένα έπρεπε κάθε υποσύνολο του σετ (*train*, *validation*, *test*) να περιέχει εικόνες ασθενών που να μην υπάρχουν σε άλλα υποσύνολα. Δηλαδή, αν για παράδειγμα υπάρχουν 5 εικόνες του ασθενή Α, τότε οι εν λόγω εικόνες πρέπει να βρίσκονται μόνο είτε στο *train* ή *validation* ή *test set* και να μην είναι διασκορπισμένες ανάμεσα στα σετ. Το εν λόγω φαινόμενο ονομάζεται *Data leakage*. Πιο συγκεκριμένα, εφόσον το μοντέλο εκπαιδευτεί σε 3 διαφορετικές εικόνες του ίδιου ασθενή που πάσχει από κορονοϊό, και οι υπόλοιπες 2 εικόνες βρίσκονται στο σετ αξιολόγησης, το μοντέλο θα μπορέσει να τις αναγνωρίσει και να βασίζεται τα αποτελέσματα του πάνω σε αυτό και όχι στα ιατρικά χαρακτηριστικά της εικόνας. Κατά συνέπεια το φαινόμενο *Data Leakage* μπορεί να προκαλέσει μεροληψία (*bias*) και ασυνέπεια στα αποτελέσματα του μοντέλου.

Όπως αναλύθηκε στο κεφάλαιο *Δεδομένα*, τα δεδομένα που χρησιμοποιήθηκαν για την εν λόγω πτυχιακή απαρτίζονται από ένα απάνθισμα σετ δεδομένων τα οποία πάρθηκαν από διάφορες πηγές. Βάση των (Maguolo & Nanni, 2020) αποδείχθηκε πως ένα μοντέλο βαθιάς μάθησης μπορεί να ξεχωρίσει εικόνες και να βασίζει την πρόβλεψη του ως προς την πηγή των εικόνων παρά ως τα ιατρικά χαρακτηριστικά τους. Στην εν λόγω πτυχιακή δεν παρατηρήθηκε να συμβαίνει κάτι τέτοιο.

Δυστυχώς, η αρχιτεκτονική της κάρτας γραφικών του υπολογιστή που χρησιμοποιήθηκε για την υλοποίηση της πτυχιακής εργασίας, κάνει την εκπαίδευση μοντέλων βαθιάς μάθησης αδύνατον. Για αυτό τον λόγο χρησιμοποιήθηκε ο επεξεργαστής *Ryzen 5600X*. Ο χρόνος εκπαίδευσης ενός μοντέλου με τη χρήση κάρτας γραφικών μειώνεται ραγδαία. Κατά συνέπεια, η εκπαίδευση κάθε μοντέλου διαρκούσε πολύ περισσότερο με αποτέλεσμα να μην υπάρχει αρκετός χρόνος για περαιτέρω τροποποιήσεις. Βέβαια, υπάρχουν λύσεις για αυτό το πρόβλημα, όπως *Cloud Computing*. Συνεπώς, έγινε δοκιμή της πλατφόρμας *Google Colab* για την εκπαίδευση των 5 μοντέλων που αναφέρθηκαν παραπάνω. Ο χρόνος εκπαίδευσης ήταν όντως εξαιρετικά μικρότερος. Ωστόσο, η πλατφόρμα ήταν ασταθής ως προς την απόδοση της, ειδικά αν ο χρήστης δεν είχε την *premium* συνδρομή. Επομένως, η υλοποίηση συνέχισε σε προσωπικό υπολογιστή.

## 9.5. Επίλυση προβλημάτων

Όπως αναφέρθηκε παραπάνω τα δεδομένα που χρησιμοποιήθηκαν απαρτίζονται από ένα απάνθισμα σετ δεδομένων, τα οποία πάρθηκαν από διάφορες πηγές. Ως κατά συνέπεια, το μέγεθος του σετ είναι επαρκές για την εκπαίδευση μοντέλων βαθιάς μάθησης με σκοπό την ανίχνευση κορονοϊού. Επίσης, το γεγονός πως το σετ απαρτίζεται από διάφορες πηγές, θεωρητικά επιτρέπει τη γενίκευση του μοντέλου σε ένα μεγαλύτερο φάσμα εικόνων.

Χρησιμοποιώντας διάφορες μεθόδους, αποφεύχθηκε η υπερπροσαρμογή (*overfitting*) των μοντέλων στο σετ δεδομένων. Συγκεκριμένα, χρησιμοποιήθηκε το (*Tf.Keras.Layers.Dropout*, n.d.) *layer* του *Tensorflow* το οποίο μηδενίζει τυχαία μερικούς νευρώνες (κατά 20% στην προκειμένη πτυχιακή) και κατά συνέπεια αναγκάζει το μοντέλο στο να μη βασίζει τη γνώση του σε συγκεκριμένους νευρώνες (Srivastava et al., 2014). Η επόμενη μέθοδος που χρησιμοποιήθηκε είναι το *Data Augmentation*, όπου στην εν λόγω μέθοδο εφαρμόζονται διάφοροι μετασχηματισμοί στην κάθε δέσμη (*batch*) του σετ δεδομένων. Συγκεκριμένα, το μοντέλο μετά από κάθε δέσμη τροφοδοτείται συνεχώς με διαφορετικές εικόνες. Αποτέλεσμα αυτού είναι η δυσκολία υπερπροσαρμογής εφόσον τα δεδομένα αλλάζουν συνεχώς. Τέλος, χρησιμοποιήθηκε το *callback Early Stopping* (*EarlyStopping*, n.d.) του *Tensorflow*, όπου σταματάει η εκπαίδευση του μοντέλου εφόσον δεν υπάρχει πρόοδος στην απώλεια του *validation set* για 8 συνεχόμενα *epochs*. Με λίγα λόγια αν η απόδοση του μοντέλου βελτιώνεται συνεχώς πάνω στο *train set* αλλά όχι στο *validation set*, τότε κατά συνέπεια το μοντέλο πάσχει από *overfitting*. Το *Early Stopping* αποτρέπει αυτό το φαινόμενο εφόσον παρακολουθείται η απόδοση στο *validation set*.

Όπως φαίνεται στο κεφάλαιο *Αποτελέσματα* κάθε μοντέλο απαρτίζεται από εκατομμύρια παραμέτρους. Η ολική εκπαίδευση τέτοιων μοντέλων σε ένα μέτριο επεξεργαστή θα έπαιρνε τεράστιο χρόνο. Ωστόσο, η χρήση *Transfer Learning* επιτρέπει στον ερευνητή τη χρήση τέτοιων μοντέλων χωρίς να απαιτείται εξειδικευμένος εξοπλισμός. Ως πείραμα, έγινε προσπάθεια εκπαίδευσης του *ResNet101* χωρίς *Transfer Learning*, όπου εκπαιδεύτηκαν όλες οι παράμετροι από την αρχή. Η ολική εκπαίδευση διάρκεσε 40 *epochs*, και ο χρόνος που χρειαζόταν κάθε *epoch* ήταν 16 ώρες. Συνεπώς, ο χρόνος εκπαίδευσης των 40 *epochs* ανέρχεται στις 640 ώρες όπου αυτό μεταφράζεται σε σχεδόν 27 ημέρες. Αδιαμφισβήτητα, η χρήση *Transfer Learning* επιτρέπει σε ερευνητές μικρής και μεγάλης κλίμακας να τρέξουν τα πειράματά τους σε *state of the art* μοντέλα.

## 9.6. Μελλοντικές προτάσεις

Στο μέλλον, σκοπός είναι η διεύρυνση της εν λόγω πτυχιακής εργασίας με περισσότερες μεθόδους για την ανίχνευση κορονοϊού. Συγκεκριμένα, θα ήταν εξαιρετικά επωφελής η χρήση *lung segmentation* και *lung localization* οι οποίες μέθοδοι προσφέρουν στο μοντέλο περισσότερη πληροφορία για εκμάθηση. Παράλληλα, σημαντικό θα ήταν η έρευνα διαφόρων μεθόδων παρεμβολής (*interpolation methods*) όσο αφορά την αλλαγή μεγέθους της κάθε εικόνας ανάλογα με τις προδιαγραφές της κάθε αρχιτεκτονικής.

Εξίσου σημαντικό είναι η εκτενής διερεύνηση και ανάλυση του κάθε μοντέλου με περισσότερες μεθόδους αξιολόγησης. Συγκεκριμένα, εφόσον τα δεδομένα τις συγκεκριμένης πτυχιακής απαρτίζονται από διάφορα

σετ δεδομένων, καλό θα ήταν να χρησιμοποιηθεί η μέθοδος των (Maguolo & Nanni, 2020) για τη διερεύνηση μεροληψίας (*bias*) του κάθε μοντέλου ως προς την πηγή του κάθε σετ.

Τέλος, θα ήταν ενδιαφέρον η αξιολόγηση της απόδοσης του κάθε μοντέλου συγκριτικά με επαγγελματίες γιατρούς, όπου μία τέτοια έρευνα θα έδειχνε αν τα μοντέλα μπορούν να συναγωνιστούν γιατρούς. Παράλληλα η ανατροφοδότηση των γιατρών όσο αφορά την απόδοση των μοντέλων, θα παρείχε μία μικρή εικόνα στο αν είναι εφικτή η χρήση τους στον πραγματικό κόσμο.

## 10. Συμπεράσματα

Αδιαμφισβήτητα, ο κορονοϊός έθεσε σε κίνδυνο τη ζωή εκατομμυρίων ανθρώπων. Η γρήγορη και αποδοτική διάγνωση ασθενών με κορονοϊό έχει καθοριστικό ρόλο στην πρόληψη της εξάπλωσης του ιού. Οι αξονικές ακτινογραφίες έχουν χαμηλό κόστος, εύκολη πρόσβαση και γρήγορα αποτελέσματα. Συμπερασματικά, ένας εξαιρετικός τρόπος διάγνωσης σε σχέση με τα *RT-PCR* τεστ και τις αξονικές τομογραφίες (*CT Scans*). Επομένως, παρουσιάστηκαν ποικίλες έρευνες με σκοπό την αυτόματη διάγνωση κορονοϊού μέσα από αξονικές ακτινογραφίες όπου έδειξαν εξαιρετική απόδοση για γρήγορη ανίχνευση και διάγνωση κορονοϊού.

Στην προκειμένη πτυχιακή εργασία χρησιμοποιήθηκαν 5 μοντέλα βαθιάς μάθησης για την ανίχνευση κορονοϊού σε αξονικές ακτινογραφίες. Το σετ δεδομένων που χρησιμοποιήθηκε, μέχρι τώρα είναι το μεγαλύτερο που υπάρχει όσο αφορά αξονικές ακτινογραφίες κορονοϊού. Συγκεκριμένα, το σετ απαρτίζεται από 33,920 αξονικές ακτινογραφίες. Όλα τα μοντέλα εκπαιδεύτηκαν σε προσωπικό υπολογιστή για συνολικά 30 με 40 *epochs* ανάλογα. Επίσης, αξιολογήθηκαν σε δεδομένα που δεν έχουν ξαναδεί έτσι ώστε να υπάρχει μία πιο καθαρή εικόνα για την απόδοση τους. Κανένα από τα 5 μοντέλα δεν παρουσίασε σημάδια υπερπροσαρμογής και όλα έφεραν ακρίβεια 92% και άνω. Συγκεκριμένα, το μοντέλο με την καλύτερη απόδοση ήταν το ***ResNet101*** με **Ακρίβεια 95.7%**, **Απώλεια 0.13**, **Precision 95.7%** και **Recall 95.6%**.

Η διάγνωση και πρόγνωση αποτελούν παρόμοια πρόκληση. Ο γιατρός διαθέτει κάποιες πληροφορίες, και θέλει να μάθει πως σχετίζονται οι εν λόγω πληροφορίες με την πραγματική κατάσταση ενός ασθενή. Δηλαδή, ποια είναι η παρούσα κατάσταση του (διάγνωση) ή ποια θα είναι στο μέλλον (πρόγνωση). Η πρόβλεψη κάθε μοντέλου είναι πιθανοτική και για αυτό τον λόγο φέρνουν μαζί τους ένα τεράστιο ρίσκο. Η αξιολόγηση ενός μοντέλου σε ανεξάρτητα εξωτερικά δεδομένα μετά την εκπαίδευση είναι ο καλύτερος τρόπος για την πραγματική εξακρίβωση της απόδοσης. Ένας σημαντικός ρόλος των μοντέλων πρόβλεψης στην Ιατρική είναι υποστήριξη λήψης αποφάσεων, για αυτό τον λόγο είναι απαιτούμενο οι προβλέψεις του μοντέλου να είναι αξιόπιστες. Συμπερασματικά, βάση τα παραπάνω αν ένα σύστημα βαθιάς μάθησης



εκπαιδευτή και αξιολογηθεί με σωστές πρακτικές και δεδομένα μπορεί να χρησιμοποιηθεί ως εργαλείο υποστήριξης λήψης αποφάσεων σε ποικίλους κλάδους της Ιατρικής.

## 11. Βιβλιογραφία

- Ackley, D. H., Hinton, G. E., & Sejnowski, T. J. (1985). A learning algorithm for boltzmann machines. *Cognitive Science*, 9(1), 147–169. [https://doi.org/10.1016/S0364-0213\(85\)80012-4](https://doi.org/10.1016/S0364-0213(85)80012-4)
- Afifi, A., Hafsa, N. E., Ali, M. A. S., Alhumam, A., & Als Salman, S. (2021). An Ensemble of Global and Local-Attention Based Convolutional Neural Networks for COVID-19 Diagnosis on Chest X-ray Images. *Symmetry*, 13(1), 113. <https://doi.org/10.3390/sym13010113>
- Aggarwal, K., Mijwil, M. M., Sonia, Al-Mistarehi, A.-H., Alomari, S., Gök, M., Alaabdin, A. M. Z., & Abdulrhman, S. H. (2022). Has the Future Started? The Current Growth of Artificial Intelligence, Machine Learning, and Deep Learning. *Iraqi Journal For Computer Science and Mathematics*, 3(1), 115–123. <https://doi.org/10.52866/ijcsm.2022.01.01.013>
- Ai, T., Yang, Z., Hou, H., Zhan, C., Chen, C., Lv, W., Tao, Q., Sun, Z., & Xia, L. (2020). Correlation of Chest CT and RT-PCR Testing for Coronavirus Disease 2019 (COVID-19) in China: A Report of 1014 Cases. *Radiology*, 296(2), E32–E40. <https://doi.org/10.1148/radiol.2020200642>
- Alghamdi, A. S., Polat, K., Alghoson, A., Alshdadi, A. A., & Abd El-Latif, A. A. (2020). A novel blood pressure estimation method based on the classification of oscillometric waveforms using machine-learning methods. *Applied Acoustics*, 164, 107279. <https://doi.org/10.1016/j.apacoust.2020.107279>
- ALIMOHAMADI, Y., SEPANDI, M., TAGHDIR, M., & HOSAMIRUDSARI, H. (2020). Determine the most common clinical symptoms in COVID-19 patients: A systematic review and meta-analysis. *Journal of Preventive Medicine and Hygiene*, 61(3), E304–E312. <https://doi.org/10.15167/2421-4248/jpmh2020.61.3.1530>
- Apoorva., P., Impana., H. C., Siri., S. L., Varshitha., M. R., & Ramesh., B. (2019). Automated Criminal Identification by Face Recognition using Open Computer Vision Classifiers. *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, 775–778. <https://doi.org/10.1109/ICCMC.2019.8819850>
- Arias-Garzón, D., Alzate-Grisales, J. A., Orozco-Arias, S., Arteaga-Arteaga, H. B., Bravo-Ortiz, M. A., Mora-Rubio, A., Saborit-Torres, J. M., Serrano, J. Á. M., de la Iglesia Vayá, M., Cardona-Morales, O., &

- Tabares-Soto, R. (2021). COVID-19 detection in X-ray images using convolutional neural networks. *Machine Learning with Applications*, 6, 100138. <https://doi.org/10.1016/j.mlwa.2021.100138>
- Asif, S., Wenhui, Y., Jin, H., & Jinhai, S. (2020). Classification of COVID-19 from Chest X-ray images using Deep Convolutional Neural Network. *2020 IEEE 6th International Conference on Computer and Communications (ICCC)*, 426–433. <https://doi.org/10.1109/ICCC51575.2020.9344870>
- Bain, A. (1873). *Mind and body The theories of their relation*. D. Appleton and Company. [http://books.google.com/books?id=eKX\\_6Y6Ab2gC](http://books.google.com/books?id=eKX_6Y6Ab2gC)
- Barua, B., Gomes, C., Baghe, S., & Sisodia, J. (2019). A Self-Driving Car Implementation using Computer Vision for Detection and Navigation. *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*, 271–274. <https://doi.org/10.1109/ICCS45141.2019.9065627>
- Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2), 157–166. <https://doi.org/10.1109/72.279181>
- BIMCV-COVID19 – BIMCV*. (n.d.). Retrieved August 25, 2022, from <https://bimcv.cipf.es/bimcv-projects/bimcv-covid19/>
- Bozinovski, S. (2020). Reminder of the First Paper on Transfer Learning in Neural Networks, 1976. *Informatica*, 44(3), Article 3. <https://doi.org/10.31449/inf.v44i3.2828>
- Brenner, D. J., & Hall, E. J. (2007). Computed Tomography—An Increasing Source of Radiation Exposure. *New England Journal of Medicine*, 357(22), 2277–2284. <https://doi.org/10.1056/NEJMra072149>
- Candemir, S., Jaeger, S., Palaniappan, K., Musco, J. P., Singh, R. K., Zhiyun Xue, null, Karargyris, A., Antani, S., Thoma, G., & McDonald, C. J. (2014). Lung segmentation in chest radiographs using anatomical atlases with nonrigid registration. *IEEE Transactions on Medical Imaging*, 33(2), 577–590. <https://doi.org/10.1109/TMI.2013.2290491>
- CategoricalAccuracy*. (n.d.). TensorFlow. Retrieved September 5, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/metrics/CategoricalAccuracy](https://www.tensorflow.org/api_docs/python/tf/keras/metrics/CategoricalAccuracy)
- CategoricalCrossentropy*. (n.d.). TensorFlow. Retrieved September 5, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/losses/CategoricalCrossentropy](https://www.tensorflow.org/api_docs/python/tf/keras/losses/CategoricalCrossentropy)
- Chen, D., Liu, S., Kingsbury, P., Sohn, S., Storlie, C. B., Habermann, E. B., Naessens, J. M., Larson, D. W., & Liu, H. (2019). Deep learning and alternative learning strategies for retrospective real-world clinical data. *Npj Digital Medicine*, 2(1), 1–5. <https://doi.org/10.1038/s41746-019-0122-0>

*Chest X-Ray Images (Pneumonia)*. (n.d.). Retrieved August 25, 2022, from <https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia>

Chollet, F. (2018). *Deep learning with Python*. Manning Publications Co.

Chowdhury, M. E. H., Rahman, T., Khandakar, A., Mazhar, R., Kadir, M. A., Mahbub, Z. B., Islam, K. R., Khan, M. S., Iqbal, A., Emadi, N. A., Reaz, M. B. I., & Islam, M. T. (2020). Can AI Help in Screening Viral and COVID-19 Pneumonia? *IEEE Access*, 8, 132665–132676. <https://doi.org/10.1109/ACCESS.2020.3010287>

*Coronavirus Graphs: Worldwide Cases and Deaths - Worldometer*. (n.d.). Retrieved June 17, 2022, from <https://www.worldometers.info/coronavirus/worldwide-graphs/#total-cases>

Courtney, J. W., Magee, M. J., & Aggarwal, J. K. (1984). Robot guidance using computer vision. *Pattern Recognition*, 17(6), 585–592. [https://doi.org/10.1016/0031-3203\(84\)90012-8](https://doi.org/10.1016/0031-3203(84)90012-8)

*COVID-19 DATABASE - SIRM*. (n.d.). Retrieved August 25, 2022, from <https://sirm.org/category/covid-19/>

*COVID-19 Radiography Database*. (n.d.). Retrieved August 25, 2022, from <https://www.kaggle.com/datasets/tawsifurrahman/covid19-radiography-database>

*Covid-19-image-repository*. (n.d.). GitHub. Retrieved August 25, 2022, from <https://github.com/ml-workgroup/covid-19-image-repository>

*COVID-QU-Ex Dataset*. (n.d.). Retrieved August 25, 2022, from <https://www.kaggle.com/datasets/cf77495622971312010dd5934ee91f07ccbcfdea8e2f7778977ea8485c1914df>

*CSVLogger*. (n.d.). TensorFlow. Retrieved September 5, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/callbacks/CSVLogger](https://www.tensorflow.org/api_docs/python/tf/keras/callbacks/CSVLogger)

Degerli, A., Ahishali, M., Yamac, M., Kiranyaz, S., Chowdhury, M. E. H., Hameed, K., Hamid, T., Mazhar, R., & Gabbouj, M. (2021). COVID-19 infection map generation and detection from chest X-ray images. *Health Information Science and Systems*, 9(1), 15. <https://doi.org/10.1007/s13755-021-00146-8>

DeGrave, A. J., Janizek, J. D., & Lee, S.-I. (2021). AI for radiographic COVID-19 detection selects shortcuts over signal. *Nature Machine Intelligence*, 3(7), 610–619. <https://doi.org/10.1038/s42256-021-00338-7>

Demšar, J. (2006). Statistical Comparisons of Classifiers over Multiple Data Sets. *Journal of Machine Learning Research*, 7(1), 1–30.

*DenseNet121*. (n.d.). TensorFlow. Retrieved September 4, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/applications/densenet/DenseNet121](https://www.tensorflow.org/api_docs/python/tf/keras/applications/densenet/DenseNet121)

*DenseNet169*. (n.d.). TensorFlow. Retrieved September 4, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/applications/densenet/DenseNet169](https://www.tensorflow.org/api_docs/python/tf/keras/applications/densenet/DenseNet169)

*Densenet.preprocess\_input*. (n.d.). Retrieved September 4, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/applications/densenet/preprocess\\_input](https://www.tensorflow.org/api_docs/python/tf/keras/applications/densenet/preprocess_input)

*Diagrams.net*. (n.d.). Retrieved September 7, 2022, from <https://app.diagrams.net/>

*EarlyStopping*. (n.d.). TensorFlow. Retrieved September 5, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/callbacks/EarlyStopping](https://www.tensorflow.org/api_docs/python/tf/keras/callbacks/EarlyStopping)

*Emerging 2019 Novel Coronavirus (2019-nCoV) Pneumonia*. (n.d.). <https://doi.org/10.1148/radiol.2020200274>

Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., & Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639), 115–118. <https://doi.org/10.1038/nature21056>

*Eurorad.org*. (n.d.). Eurorad - Brought to You by the ESR. Retrieved August 25, 2022, from <https://www.eurorad.org/homepage>

*FalseNegatives*. (n.d.). TensorFlow. Retrieved September 5, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/metrics/FalseNegatives](https://www.tensorflow.org/api_docs/python/tf/keras/metrics/FalseNegatives)

*FalsePositives*. (n.d.). TensorFlow. Retrieved September 5, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/metrics/FalsePositives](https://www.tensorflow.org/api_docs/python/tf/keras/metrics/FalsePositives)

Federoff, H. J., & Gostin, L. O. (2009). Evolving From Reductionism to Holism: Is There a Future for Systems Medicine? *JAMA*, 302(9), 994–996. <https://doi.org/10.1001/jama.2009.1264>

Flores, M., Glusman, G., Brogaard, K., Price, N. D., & Hood, L. (2013). P4 medicine: How systems medicine will transform the healthcare sector and society. *Personalized Medicine*, 10(6), 565–576. <https://doi.org/10.2217/pme.13.57>

Gao, J., Yang, Y., Lin, P., & Park, D. S. (2018). Computer Vision in Healthcare Applications. *Journal of Healthcare Engineering*, 2018, e5157020. <https://doi.org/10.1155/2018/5157020>

- Geitung, J. T., Skjaerstad, L. M., & Göthlin, J. H. (1999). Clinical utility of chest roentgenograms. *European Radiology*, 9(4), 721–723. <https://doi.org/10.1007/s003300050741>
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
- Gulshan, V., Peng, L., Coram, M., Stumpe, M. C., Wu, D., Narayanaswamy, A., Venugopalan, S., Widner, K., Madams, T., Cuadros, J., Kim, R., Raman, R., Nelson, P. C., Mega, J. L., & Webster, D. R. (2016). Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathy in Retinal Fundus Photographs. *JAMA*, 316(22), 2402–2410. <https://doi.org/10.1001/jama.2016.17216>
- Haggag, N. T., Sedik, A., Elbanby, G. M., El-Fishawy, A. S., Dessouky, M. I., & Khalaf, A. A. M. (2019). Classification of Corneal Pattern Based on Convolutional LSTM Neural Network. *Menoufia Journal of Electronic Engineering Research*, 28(ICEEM2019-Special Issue), 158–162. <https://doi.org/10.21608/mjeer.2019.76998>
- Haghanifar, A. (2022). *COVID-CXNet* [Jupyter Notebook]. <https://github.com/armiro/COVID-CXNet> (Original work published 2020)
- Halevy, A., Norvig, P., & Pereira, F. (2009). The Unreasonable Effectiveness of Data. *IEEE Intelligent Systems*, 24(2), 8–12. <https://doi.org/10.1109/MIS.2009.36>
- Hammad, M., Maher, A., Wang, K., Jiang, F., & Amrani, M. (2018). Detection of abnormal heart conditions based on characteristics of ECG signals. *Measurement*, 125, 634–644. <https://doi.org/10.1016/j.measurement.2018.05.033>
- He, K., Zhang, X., Ren, S., & Sun, J. (2015a). *Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification* (arXiv:1502.01852). arXiv. <http://arxiv.org/abs/1502.01852>
- He, K., Zhang, X., Ren, S., & Sun, J. (2015b). *Deep Residual Learning for Image Recognition* (arXiv:1512.03385). arXiv. <https://doi.org/10.48550/arXiv.1512.03385>
- Hebb, D. O. (1949). *The organization of behavior; a neuropsychological theory*. (pp. xix, 335). Wiley.
- Hinton, G. E., Osindero, S., & Teh, Y.-W. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7), 1527–1554. <https://doi.org/10.1162/neco.2006.18.7.1527>
- Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8), 2554–2558. <https://doi.org/10.1073/pnas.79.8.2554>

- Huang, C., Wang, Y., Li, X., Ren, L., Zhao, J., Hu, Y., Zhang, L., Fan, G., Xu, J., Gu, X., Cheng, Z., Yu, T., Xia, J., Wei, Y., Wu, W., Xie, X., Yin, W., Li, H., Liu, M., ... Cao, B. (2020). Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China. *The Lancet*, 395(10223), 497–506. [https://doi.org/10.1016/S0140-6736\(20\)30183-5](https://doi.org/10.1016/S0140-6736(20)30183-5)
- Huang, G., Liu, Z., van der Maaten, L., & Weinberger, K. Q. (2018). *Densely Connected Convolutional Networks* (arXiv:1608.06993). arXiv. <https://doi.org/10.48550/arXiv.1608.06993>
- Iftner, T., Iftner, A., Pohle, D., & Martus, P. (2022). *Evaluation of the specificity and accuracy of SARS-CoV-2 rapid antigen self-tests compared to RT-PCR from 1015 asymptomatic volunteers* (p. 2022.02.11.22270873). medRxiv. <https://doi.org/10.1101/2022.02.11.22270873>
- InceptionV3*. (n.d.). TensorFlow. Retrieved September 3, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/applications/inception\\_v3/InceptionV3](https://www.tensorflow.org/api_docs/python/tf/keras/applications/inception_v3/InceptionV3)
- Inception\_v3.preprocess\_input*. (n.d.). Retrieved September 4, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/applications/inception\\_v3/preprocess\\_input](https://www.tensorflow.org/api_docs/python/tf/keras/applications/inception_v3/preprocess_input)
- Iqbal, M. J., Iqbal, M. M., Ahmad, I., Alassafi, M. O., Alfakeeh, A. S., & Alhomoud, A. (2021). Real-Time Surveillance Using Deep Learning. *Security and Communication Networks*, 2021, e6184756. <https://doi.org/10.1155/2021/6184756>
- Jaeger, S., Karargyris, A., Candemir, S., Folio, L., Siegelman, J., Callaghan, F., Zhiyun Xue, null, Palaniappan, K., Singh, R. K., Antani, S., Thoma, G., Yi-Xiang Wang, null, Pu-Xuan Lu, null, & McDonald, C. J. (2014). Automatic tuberculosis screening using chest radiographs. *IEEE Transactions on Medical Imaging*, 33(2), 233–245. <https://doi.org/10.1109/TMI.2013.2284099>
- Jain, R., Gupta, M., Taneja, S., & Hemanth, D. J. (2021). Deep learning based detection and analysis of COVID-19 on chest X-ray images. *Applied Intelligence*, 51(3), 1690–1700. <https://doi.org/10.1007/s10489-020-01902-1>
- Kamilaris, A., & Prenafeta-Boldú, F. X. (2018). A review of the use of convolutional neural networks in agriculture. *The Journal of Agricultural Science*, 156(3), 312–322. <https://doi.org/10.1017/S0021859618000436>
- Keidar, D., Yaron, D., Goldstein, E., Shachar, Y., Blass, A., Charbinsky, L., Aharon, I., Lifshitz, L., Lumelsky, D., Neeman, Z., Mizrahi, M., Hajouj, M., Eizenbach, N., Sela, E., Weiss, C. S., Levin, P., Benjaminov, O., Bachar, G. N., Tamir, S., ... Eldar, Y. C. (2021). COVID-19 classification of X-ray images

using deep neural networks. *European Radiology*, 31(12), 9654–9663. <https://doi.org/10.1007/s00330-021-08050-1>

*Keras*. (n.d.). Retrieved August 29, 2022, from <https://keras.io/>

Khalil, H., El-Hag, N., Sedik, A., El-Shafie, W., Mohamed, A. E.-N., Khalaf, A. A. M., El-Banby, G. M., Abd El-Samie, F. I., & El-Fishawy, A. S. (2019). Classification of Diabetic Retinopathy types based on Convolution Neural Network (CNN). *Menoufia Journal of Electronic Engineering Research*, 28(ICEEM2019-Special Issue), 126–153. <https://doi.org/10.21608/mjeer.2019.76962>

Khan, W., Zaki, N., & Ali, L. (2021). Intelligent Pneumonia Identification From Chest X-Rays: A Systematic Literature Review. *IEEE Access*, 9, 51747–51771. <https://doi.org/10.1109/ACCESS.2021.3069937>

Kieseberg, P., Hobel, H., Schrittwieser, S., Weippl, E., & Holzinger, A. (2014). Protecting Anonymity in Data-Driven Biomedical Science. In A. Holzinger & I. Jurisica (Eds.), *Interactive Knowledge Discovery and Data Mining in Biomedical Informatics: State-of-the-Art and Future Challenges* (pp. 301–316). Springer. [https://doi.org/10.1007/978-3-662-43968-5\\_17](https://doi.org/10.1007/978-3-662-43968-5_17)

Kingma, D. P., & Ba, J. (2017). *Adam: A Method for Stochastic Optimization* (arXiv:1412.6980). arXiv. <http://arxiv.org/abs/1412.6980>

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems*, 25. <https://papers.nips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html>

Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278–2324. <https://doi.org/10.1109/5.726791>

Lee, H., Grosse, R., Ranganath, R., & Ng, A. Y. (2009). Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. *Proceedings of the 26th Annual International Conference on Machine Learning*, 609–616. <https://doi.org/10.1145/1553374.1553453>

Liang, M., Li, Z., Chen, T., & Zeng, J. (2015). Integrative Data Analysis of Multi-Platform Cancer Data with a Multimodal Deep Learning Approach. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 12(4), 928–937. <https://doi.org/10.1109/TCBB.2014.2377729>

Lin, M., Chen, Q., & Yan, S. (2014). *Network In Network* (arXiv:1312.4400). arXiv. <http://arxiv.org/abs/1312.4400>

Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollar, P. (2017). *Focal Loss for Dense Object Detection*. 2980–2988.

[https://openaccess.thecvf.com/content\\_iccv\\_2017/html/Lin\\_Focal\\_Loss\\_for\\_ICCV\\_2017\\_paper.html](https://openaccess.thecvf.com/content_iccv_2017/html/Lin_Focal_Loss_for_ICCV_2017_paper.html)

Lin, Z. Q., Shafiee, M. J., Bochkarev, S., Jules, M. S., Wang, X. Y., & Wong, A. (2019). *Do Explanations Reflect Decisions? A Machine-centric Strategy to Quantify the Performance of Explainability Algorithms* (arXiv:1910.07387). arXiv. <https://doi.org/10.48550/arXiv.1910.07387>

Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., van der Laak, J. A. W. M., van Ginneken, B., & Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42, 60–88. <https://doi.org/10.1016/j.media.2017.07.005>

Liu, F., Yadav, P., Baschnagel, A. M., & McMillan, A. B. (2019). MR-based treatment planning in radiation therapy using a deep learning approach. *Journal of Applied Clinical Medical Physics*, 20(3), 105–114. <https://doi.org/10.1002/acm2.12554>

Lou, B., Doken, S., Zhuang, T., Wingerter, D., Gidwani, M., Mistry, N., Ladic, L., Kamen, A., & Abazeed, M. E. (2019). An image-based deep learning framework for individualising radiotherapy dose: A retrospective analysis of outcome prediction. *The Lancet Digital Health*, 1(3), e136–e147. [https://doi.org/10.1016/S2589-7500\(19\)30058-5](https://doi.org/10.1016/S2589-7500(19)30058-5)

Madani, A., Arnaout, R., Mofrad, M., & Arnaout, R. (2018). Fast and accurate view classification of echocardiograms using deep learning. *Npj Digital Medicine*, 1(1), 1–8. <https://doi.org/10.1038/s41746-017-0013-1>

Maglogiannis, I., & Doukas, C. N. (2009). Overview of Advanced Computer Vision Systems for Skin Lesions Characterization. *IEEE Transactions on Information Technology in Biomedicine*, 13(5), 721–733. <https://doi.org/10.1109/TITB.2009.2017529>

Maguolo, G., & Nanni, L. (2020). *A Critic Evaluation of Methods for COVID-19 Automatic Detection from X-Ray Images* (arXiv:2004.12823). arXiv. <https://doi.org/10.48550/arXiv.2004.12823>

Mandelbaum, E. (2020). Associationist Theories of Thought. In E. N. Zalta (Ed.), *The Stanford Encyclopedia of Philosophy* (Fall 2020). Metaphysics Research Lab, Stanford University. <https://plato.stanford.edu/archives/fall2020/entries/associationist-thought/>

*Mariosconsta/thesis*. (n.d.). GitHub. Retrieved September 13, 2022, from [https://github.com/mariosconsta/bcs\\_thesis](https://github.com/mariosconsta/bcs_thesis)

*Matplotlib*. (n.d.). Retrieved August 29, 2022, from <https://matplotlib.org/>



McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4), 115–133. <https://doi.org/10.1007/BF02478259>

*Microsoft Excel*. (n.d.). Retrieved September 7, 2022, from <https://www.microsoft.com/en-us/microsoft-365/excel>

*Microsoft Word*. (n.d.). Retrieved September 7, 2022, from <https://www.microsoft.com/en-us/microsoft-365/word>

*ModelCheckpoint*. (n.d.). TensorFlow. Retrieved September 5, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/callbacks/ModelCheckpoint](https://www.tensorflow.org/api_docs/python/tf/keras/callbacks/ModelCheckpoint)

Narin, A., Kaya, C., & Pamuk, Z. (2021). Automatic detection of coronavirus disease (COVID-19) using X-ray images and deep convolutional neural networks. *Pattern Analysis and Applications*, 24(3), 1207–1220. <https://doi.org/10.1007/s10044-021-00984-y>

Nemenyi, P. (1963). *Distribution-free Multiple Comparisons*. Princeton University.

*NumPy*. (n.d.). Retrieved August 29, 2022, from <https://numpy.org/>

*PADCHEST – BIMCV*. (n.d.). Retrieved August 25, 2022, from <https://bimcv.cipf.es/bimcv-projects/padchest/>

*Pandas*. (n.d.). Retrieved August 29, 2022, from <https://pandas.pydata.org/>

PhD, J. P. C. (2022). *Ieee8023/covid-chestxray-dataset* [Jupyter Notebook]. <https://github.com/ieee8023/covid-chestxray-dataset> (Original work published 2020)

*Precision*. (n.d.). TensorFlow. Retrieved September 5, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/metrics/Precision](https://www.tensorflow.org/api_docs/python/tf/keras/metrics/Precision)

*Project Jupyter*. (n.d.). Retrieved August 30, 2022, from <https://jupyter.org>

*Python*. (n.d.). Python.Org. Retrieved August 29, 2022, from <https://www.python.org/>

Qaid, T. S., Mazaar, H., Al-Shamri, M. Y. H., Alqahtani, M. S., Raweh, A. A., & Alakwaa, W. (2021). Hybrid Deep-Learning and Machine-Learning Models for Predicting COVID-19. *Computational Intelligence and Neuroscience*, 2021, e9996737. <https://doi.org/10.1155/2021/9996737>

Rahman, T., Khandakar, A., Qiblawey, Y., Tahir, A., Kiranyaz, S., Abul Kashem, S. B., Islam, M. T., Al Maadeed, S., Zughaier, S. M., Khan, M. S., & Chowdhury, M. E. H. (2021). Exploring the effect of image

enhancement techniques on COVID-19 detection using chest X-ray images. *Computers in Biology and Medicine*, 132, 104319. <https://doi.org/10.1016/j.combiomed.2021.104319>

Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., Duan, T., Ding, D., Bagul, A., Langlotz, C., Shpanskaya, K., Lungren, M. P., & Ng, A. Y. (2017). *CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning* (arXiv:1711.05225). arXiv. <https://doi.org/10.48550/arXiv.1711.05225>

*Recall*. (n.d.). TensorFlow. Retrieved September 5, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/metrics/Recall](https://www.tensorflow.org/api_docs/python/tf/keras/metrics/Recall)

*ReduceLROnPlateau*. (n.d.). TensorFlow. Retrieved September 5, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/callbacks/ReduceLROnPlateau](https://www.tensorflow.org/api_docs/python/tf/keras/callbacks/ReduceLROnPlateau)

Resler, A., Yeshurun, R., Natalio, F., & Giryas, R. (2021). A deep-learning model for predictive archaeology and archaeological community detection. *Humanities and Social Sciences Communications*, 8(1), 1–10. <https://doi.org/10.1057/s41599-021-00970-z>

*ResNet50V2*. (n.d.). TensorFlow. Retrieved September 4, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/applications/resnet\\_v2/ResNet50V2](https://www.tensorflow.org/api_docs/python/tf/keras/applications/resnet_v2/ResNet50V2)

*ResNet101V2*. (n.d.). TensorFlow. Retrieved September 4, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/applications/resnet\\_v2/ResNet101V2](https://www.tensorflow.org/api_docs/python/tf/keras/applications/resnet_v2/ResNet101V2)

*Resnet\_v2.preprocess\_input*. (n.d.). Retrieved September 4, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/applications/resnet\\_v2/preprocess\\_input](https://www.tensorflow.org/api_docs/python/tf/keras/applications/resnet_v2/preprocess_input)

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. In N. Navab, J. Hornegger, W. M. Wells, & A. F. Frangi (Eds.), *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015* (pp. 234–241). Springer International Publishing. [https://doi.org/10.1007/978-3-319-24574-4\\_28](https://doi.org/10.1007/978-3-319-24574-4_28)

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6), 386–408. <https://doi.org/10.1037/h0042519>

*RSNA Pneumonia Detection Challenge*. (n.d.). Retrieved August 25, 2022, from <https://kaggle.com/competitions/rsna-pneumonia-detection-challenge>

Salakhutdinov, R., & Hinton, G. (2009). Deep Boltzmann Machines. *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*, 448–455. <https://proceedings.mlr.press/v5/salakhutdinov09a.html>

*Scikit-learn*. (n.d.). Retrieved August 29, 2022, from <https://scikit-learn.org/stable/>

*Seaborn*. (n.d.). Retrieved August 29, 2022, from <https://seaborn.pydata.org/>

*Seaborn.heatmap*. (n.d.). Retrieved September 7, 2022, from <https://seaborn.pydata.org/generated/seaborn.heatmap.html>

Shi, F., Wang, J., Shi, J., Wu, Z., Wang, Q., Tang, Z., He, K., Shi, Y., & Shen, D. (2021). Review of Artificial Intelligence Techniques in Imaging Data Acquisition, Segmentation, and Diagnosis for COVID-19. *IEEE Reviews in Biomedical Engineering*, 14, 4–15. <https://doi.org/10.1109/RBME.2020.2987975>

Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*, 6(1), 60. <https://doi.org/10.1186/s40537-019-0197-0>

Simonyan, K., & Zisserman, A. (2015). *Very Deep Convolutional Networks for Large-Scale Image Recognition* (arXiv:1409.1556). arXiv. <https://doi.org/10.48550/arXiv.1409.1556>

*Sklearn metrics classification report*. (n.d.). Scikit-Learn. Retrieved August 30, 2022, from [https://scikit-learn.org/stable/modules/generated/sklearn.metrics.classification\\_report.html](https://scikit-learn.org/stable/modules/generated/sklearn.metrics.classification_report.html)

*Sklearn metrics confusion matrix*. (n.d.). Scikit-Learn. Retrieved August 30, 2022, from [https://scikit-learn.org/stable/modules/generated/sklearn.metrics.confusion\\_matrix.html](https://scikit-learn.org/stable/modules/generated/sklearn.metrics.confusion_matrix.html)

Smolensky, P. (1986). *Information Processing in Dynamical Systems: Foundations of Harmony Theory*. COLORADO UNIV AT BOULDER DEPT OF COMPUTER SCIENCE. <https://apps.dtic.mil/sti/citations/ADA620727>

Speets, A. M., van der Graaf, Y., Hoes, A. W., Kalmijn, S., Sachs, A. P., Rutten, M. J., Gratama, J. W. C., Montauban van Swijndregt, A. D., & Mali, W. Pt. (2006). Chest radiography in general practice: Indications, diagnostic yield and consequences for patient management. *The British Journal of General Practice*, 56(529), 574–578.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15(56), 1929–1958.

- Sun, C., Shrivastava, A., Singh, S., & Gupta, A. (2017). *Revisiting Unreasonable Effectiveness of Data in Deep Learning Era*. 843–852. [https://openaccess.thecvf.com/content\\_iccv\\_2017/html/Sun\\_Revisiting\\_Unreasonable\\_Effectiveness\\_ICCV\\_2017\\_paper.html](https://openaccess.thecvf.com/content_iccv_2017/html/Sun_Revisiting_Unreasonable_Effectiveness_ICCV_2017_paper.html)
- Suresh, H., Hunt, N., Johnson, A., Celi, L. A., Szolovits, P., & Ghassemi, M. (2017). *Clinical Intervention Prediction and Understanding using Deep Networks* (arXiv:1705.08498). arXiv. <https://doi.org/10.48550/arXiv.1705.08498>
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2014). *Going Deeper with Convolutions* (arXiv:1409.4842). arXiv. <http://arxiv.org/abs/1409.4842>
- Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2015). *Rethinking the Inception Architecture for Computer Vision* (arXiv:1512.00567). arXiv. <https://doi.org/10.48550/arXiv.1512.00567>
- Tahamtan, A., & Ardebili, A. (2020). Real-time RT-PCR in COVID-19 detection: Issues affecting the results. *Expert Review of Molecular Diagnostics*, 20(5), 453–454. <https://doi.org/10.1080/14737159.2020.1757437>
- Tahir, A. M., Chowdhury, M. E. H., Khandakar, A., Rahman, T., Qiblawey, Y., Khurshid, U., Kiranyaz, S., Ibtehaz, N., Rahman, M. S., Al-Maadeed, S., Mahmud, S., Ezeddin, M., Hameed, K., & Hamid, T. (2021). COVID-19 infection localization and severity grading from chest X-ray images. *Computers in Biology and Medicine*, 139, 105002. <https://doi.org/10.1016/j.combiomed.2021.105002>
- Tai, M. C.-T. (2020). The impact of artificial intelligence on human society and bioethics. *Tzu-Chi Medical Journal*, 32(4), 339–343. [https://doi.org/10.4103/tcmj.tcmj\\_71\\_20](https://doi.org/10.4103/tcmj.tcmj_71_20)
- TensorBoard*. (n.d.). Retrieved September 5, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/callbacks/TensorBoard](https://www.tensorflow.org/api_docs/python/tf/keras/callbacks/TensorBoard)
- TensorFlow*. (n.d.). Retrieved August 29, 2022, from <https://www.tensorflow.org/>
- Tf.keras.Input*. (n.d.). TensorFlow. Retrieved September 4, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/Input](https://www.tensorflow.org/api_docs/python/tf/keras/Input)
- Tf.keras.layers.Dense*. (n.d.). TensorFlow. Retrieved September 4, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/layers/Dense](https://www.tensorflow.org/api_docs/python/tf/keras/layers/Dense)

*Tf.keras.layers.Dropout*. (n.d.). TensorFlow. Retrieved September 4, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/layers/Dropout](https://www.tensorflow.org/api_docs/python/tf/keras/layers/Dropout)

*Tf.keras.layers.Flatten*. (n.d.). Retrieved September 5, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/layers/Flatten](https://www.tensorflow.org/api_docs/python/tf/keras/layers/Flatten)

*Tf.keras.layers.GlobalAveragePooling2D*. (n.d.). Retrieved September 4, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/layers/GlobalAveragePooling2D](https://www.tensorflow.org/api_docs/python/tf/keras/layers/GlobalAveragePooling2D)

*Tf.keras.layers.RandomFlip*. (n.d.). TensorFlow. Retrieved September 1, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/layers/RandomFlip](https://www.tensorflow.org/api_docs/python/tf/keras/layers/RandomFlip)

*Tf.keras.layers.RandomRotation*. (n.d.). TensorFlow. Retrieved September 1, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/layers/RandomRotation](https://www.tensorflow.org/api_docs/python/tf/keras/layers/RandomRotation)

*Tf.keras.Model.evaluate*. (n.d.). TensorFlow. Retrieved September 11, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/Model#evaluate](https://www.tensorflow.org/api_docs/python/tf/keras/Model#evaluate)

*Tf.keras.utils.image\_dataset\_from\_directory*. (n.d.). TensorFlow. Retrieved September 1, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/utils/image\\_dataset\\_from\\_directory](https://www.tensorflow.org/api_docs/python/tf/keras/utils/image_dataset_from_directory)

Toğaçar, M., Ergen, B., & Cömert, Z. (2020). COVID-19 detection using deep learning models to exploit Social Mimic Optimization and structured chest X-ray images using fuzzy color and stacking approaches. *Computers in Biology and Medicine*, 121, 103805. <https://doi.org/10.1016/j.compbimed.2020.103805>

*TrueNegatives*. (n.d.). TensorFlow. Retrieved September 5, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/metrics/TrueNegatives](https://www.tensorflow.org/api_docs/python/tf/keras/metrics/TrueNegatives)

*TruePositives*. (n.d.). TensorFlow. Retrieved September 5, 2022, from [https://www.tensorflow.org/api\\_docs/python/tf/keras/metrics/TruePositives](https://www.tensorflow.org/api_docs/python/tf/keras/metrics/TruePositives)

*Visual Studio Code—Code Editing. Redefined*. (n.d.). Retrieved August 29, 2022, from <https://code.visualstudio.com/>

Wang, H., Pujos-Guillot, E., Comte, B., de Miranda, J. L., Spiwok, V., Chorbev, I., Castiglione, F., Tieri, P., Watterson, S., McAllister, R., de Melo Malaquias, T., Zanin, M., Rai, T. S., & Zheng, H. (2021). Deep learning in systems medicine. *Briefings in Bioinformatics*, 22(2), 1543–1559. <https://doi.org/10.1093/bib/bbaa237>

Wang, H., & Raj, B. (2017). *On the Origin of Deep Learning* (arXiv:1702.07800). arXiv. <http://arxiv.org/abs/1702.07800>

- Wang, J., Ma, Y., Zhang, L., Gao, R. X., & Wu, D. (2018). Deep learning for smart manufacturing: Methods and applications. *Journal of Manufacturing Systems*, 48, 144–156. <https://doi.org/10.1016/j.jmsy.2018.01.003>
- Wang, L. (2022). *COVID-Net Open Source Initiative* [Jupyter Notebook]. <https://github.com/lindawangg/COVID-Net> (Original work published 2020)
- Wang, L., Lin, Z. Q., & Wong, A. (2020). COVID-Net: A tailored deep convolutional neural network design for detection of COVID-19 cases from chest X-ray images. *Scientific Reports*, 10(1), 19549. <https://doi.org/10.1038/s41598-020-76550-z>
- Wang, Y., Kang, H., Liu, X., & Tong, Z. (2020). Combination of RT-qPCR testing and clinical features for diagnosis of COVID-19 facilitates management of SARS-CoV-2 outbreak. *Journal of Medical Virology*, 92(6), 538–539. <https://doi.org/10.1002/jmv.25721>
- Weiss, K., Khoshgoftaar, T. M., & Wang, D. (2016). A survey of transfer learning. *Journal of Big Data*, 3(1), 9. <https://doi.org/10.1186/s40537-016-0043-6>
- WHO Director-General's opening remarks at the media briefing on COVID-19—11 March 2020. (n.d.). Retrieved June 17, 2022, from <https://www.who.int/director-general/speeches/detail/who-director-general-s-opening-remarks-at-the-media-briefing-on-covid-19---11-march-2020>
- Yasin, R., & Gouda, W. (2020). Chest X-ray findings monitoring COVID-19 disease course and severity. *Egyptian Journal of Radiology and Nuclear Medicine*, 51(1), 193. <https://doi.org/10.1186/s43055-020-00296-x>
- Yu, F., Wang, D., Shelhamer, E., & Darrell, T. (2018). Deep Layer Aggregation. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2403–2412. <https://doi.org/10.1109/CVPR.2018.00255>
- Zahoor, S., Lali, I. U., Khan, M. A., Javed, K., & Mehmood, W. (2020). Breast Cancer Detection and Classification using Traditional Computer Vision Techniques: A Comprehensive Review. *Current Medical Imaging Reviews*, 16(10), 1187–1200. <https://doi.org/10.2174/1573405616666200406110547>
- Zanin, M., Chorbev, I., Stres, B., Stalidzans, E., Vera, J., Tieri, P., Castiglione, F., Groen, D., Zheng, H., Baumbach, J., Schmid, J. A., Basilio, J., Klimek, P., Debeljak, N., Rozman, D., & Schmidt, H. H. H. W. (2019). Community effort endorsing multiscale modelling, multiscale data science and multiscale computing for systems medicine. *Briefings in Bioinformatics*, 20(3), 1057–1062. <https://doi.org/10.1093/bib/bbx160>

Zeiler, M. D., & Fergus, R. (2014). Visualizing and Understanding Convolutional Networks. In D. Fleet, T. Pajdla, B. Schiele, & T. Tuytelaars (Eds.), *Computer Vision – ECCV 2014* (pp. 818–833). Springer International Publishing. [https://doi.org/10.1007/978-3-319-10590-1\\_53](https://doi.org/10.1007/978-3-319-10590-1_53)

Zhou, Z., Rahman Siddiquee, M. M., Tajbakhsh, N., & Liang, J. (2018). UNet++: A Nested U-Net Architecture for Medical Image Segmentation. In D. Stoyanov, Z. Taylor, G. Carneiro, T. Syeda-Mahmood, A. Martel, L. Maier-Hein, J. M. R. S. Tavares, A. Bradley, J. P. Papa, V. Belagiannis, J. C. Nascimento, Z. Lu, S. Conjeti, M. Moradi, H. Greenspan, & A. Madabhushi (Eds.), *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support* (pp. 3–11). Springer International Publishing. [https://doi.org/10.1007/978-3-030-00889-5\\_1](https://doi.org/10.1007/978-3-030-00889-5_1)