ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ

ΤΜΗΜΑ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

“ΠΛΗΡΟΦΟΡΙΑΚΑ ΣΥΣΤΗΜΑΤΑ & ΥΠΗΡΕΣΙΕΣ”

Ανίχνευση μορφολογικών χαρακτηριστικών βλαστοκύστεων με χρήση μοντέλων σημασιολογικής τμηματοποίησης

Από

Τομαρά Ηλία

Υποβάλλεται

για την εκπλήρωση των προϋποθέσεων λήψης

Μεταπτυχιακού Διπλώματος

στην ειδίκευση «ΜΔΑ»

του ΠΜΣ “Πληροφοριακά Συστήματα & Υπηρεσίες”

στο

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

Απρίλιος 2025

Επιβλέπων: Μαγκλογιάννης Ηλίας

Ακαδημαϊκή Θέση: Καθηγητής

Συγγραφέας / Author: Τομαράς Ηλίας

ΣΕΛΙΔΑ ΕΓΚΥΡΟΤΗΤΑΣ

Ονοματεπώνυμο Φοιτητή: Τομαράς Ηλίας

Τίτλος Μεταπτυχιακής Διπλωματικής Εργασίας: Ανίχνευση μορφολογικών χαρακτηριστικών βλαστοκύστεων με χρήση μοντέλων σημασιολογικής τμηματοποίησης

Η παρούσα Μεταπτυχιακή Διπλωματική Εργασία υποβάλλεται ως μερική εκπλήρωση των απαιτήσεων του Προγράμματος Μεταπτυχιακών Σπουδών "Πληροφοριακά Συστήματα & Υπηρεσίες" του Τμήματος Ψηφιακών Συστημάτων του Πανεπιστημίου Πειραιώς και εγκρίθηκε στις 02/04/2025 από τα μέλη της Εξεταστικής Επιτροπής.

Εξεταστική Επιτροπή

Επιβλέπων (Τμήμα Ψηφιακών Συστημάτων, Πανεπιστήμιο Πειραιώς): Μαγκλογιάννης Ηλίας, Καθηγητής

Μέλος Εξεταστικής Επιτροπής: Μενύχτας Ανδρέας, Επίκουρος Καθηγητής

Μέλος Εξεταστικής Επιτροπής: Τελέλης Ορέστης, Επίκουρος Καθηγητής

ΥΠΕΥΘΥΝΗ ΔΗΛΩΣΗ ΑΥΘΕΝΤΙΚΟΤΗΤΑΣ

Ο Τομαράς Ηλίας, γνωρίζοντας τις συνέπειες της λογοκλοπής, δηλώνω υπεύθυνα ότι η παρούσα εργασία με τίτλο «Ανίχνευση μορφολογικών χαρακτηριστικών βλαστοκύστεων με χρήση μοντέλων σημασιολογικής τμηματοποίησης», αποτελεί προϊόν αυστηρά προσωπικής εργασίας και όλες οι πηγές που έχω χρησιμοποιήσει, έχουν δηλωθεί κατάλληλα στις βιβλιογραφικές παραπομπές και αναφορές. Τα σημεία όπου έχω χρησιμοποιήσει ιδέες, κείμενο ή/και πηγές άλλων συγγραφέων, αναφέρονται ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή. Επιπλέον δηλώνω υπεύθυνα ότι η συγκεκριμένη Μεταπτυχιακή Διπλωματική Εργασία έχει συγγραφεί από εμένα προσωπικά και δεν έχει υποβληθεί ούτε έχει αξιολογηθεί στο πλαίσιο κάποιου άλλου μεταπτυχιακού ή προπτυχιακού τίτλου σπουδών, στην Ελλάδα ή στο εξωτερικό.

Παράβαση της ανωτέρω ακαδημαϊκής μου ευθύνης αποτελεί ουσιώδη λόγο για την ανάκληση του πτυχίου μου. Σε κάθε περίπτωση, αναληθούς ή ανακριβούς δηλώσεως, υπόκειμαι στις συνέπειες που προβλέπονται τις διατάξεις που προβλέπει η Ελληνική και Κοινοτική Νομοθεσία περί πνευματικής ιδιοκτησίας.

Ο ΔΗΛΩΝ

Ονοματεπώνυμο: Τομαράς Ηλίας

Αριθμός Μητρώου: me2329

Υπογραφή:

Ευχαριστίες

Η ολοκλήρωση της παρούσας Διπλωματικής εργασίας συνδέεται παράλληλα με την ολοκλήρωση των σπουδών μου στο Π.Μ.Σ. Πληροφοριακά Συστήματα και Υπηρεσίες του Τμήματος Ψηφιακών Συστημάτων στο Πανεπιστήμιο Πειραιώς και την έναρξη της επαγγελματικής μου πορείας στον κλάδο της Επιστήμης Δεδομένων, η οποία ελπίζω να μου προσφέρει αρκετές νέες εμπειρίες, γνώσεις, εξέλιξη και ακόμη περισσότερη αγάπη και πάθος για το συγκεκριμένο αντικείμενο.

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή Μαγκλογιάννη Ηλία για την καθοδήγηση, τις συμβουλές του αλλά και την έμπνευση που μου πρόσφερε κατά την διάρκεια αυτής της διπλωματικής εργασίας.

Τέλος θα ήθελα να ευχαριστήσω θερμά τόσο τους γονείς μου όσο και την αδερφή μου για την υποστήριξη, την ενθάρρυνση αλλά και την αγάπη τους σε όλη την διάρκεια των σπουδών μου.

Περιεχόμενα

Λίστα Εικόνων

Λίστα Πινάκων

Ακρωνύμια

Ακρωνύμιο	**Περιγραφή**
ANN	Artificial Neural Network
ASPP	Atrous Spatial Pyramid Pooling
BCE	Binary Cross Entropy
CCE	Categorical Cross Entropy
CNN	Convolutional Neural Network
CRF	Conditional Random Field
FN	False Negative
FP	False Positive
FCN	Fully Convolutional Network
FNN	Feed-forward Neural Network
GRU	Gated Recurrent Unit
ICM	Inner Cell Mass
IVF	In Vitro Fertilization
IoU	Intersection of Union
LSTM	Long Short-Term Memory
PGS	Pre-implantation Genetic Screening
PPM	Pyramid Pooling Module
ReLU	Rectified Linear Unit
RGB	Red Green Blue
RNN	Recurrent Neural Network
SGD	Stochastic Gradient Descent
TE	Trophectoderm
TN	True Negative
TP	True Positive
TPR	True Positive Rate
ZP	Zona Pellucida

Περίληψη

Φιλοδοξία της παρούσας Διπλωματικής εργασίας είναι η ανάπτυξη και η σύγκριση τριών μοντέλων σημασιολογικής τμηματοποίησης, UNet, DeepLab και PSPNet, για την ανίχνευση τριών μορφολογικών χαρακτηριστικών (ICM, TE και ZP) από εικόνες βλαστοκύστεων, με σκοπό την ανάδειξη της καταλληλότητας των εμβρύων για εξωσωματική γονιμοποίηση (IVF). Τα εν λόγω μοντέλα παρουσιάζουν διαφορετική συμπεριφορά ως προς το σύνολο δεδομένων, μέσω του οποίου πραγματοποιείται η εκπαίδευση και η αξιολόγησή τους.

Στην παρούσα Διπλωματική εργασία περιγράφονται η αξία της τμηματοποίησης εικόνων στη μηχανική μάθηση και ειδικότερα στον επιστημονικό κλάδο της Ιατρικής, καθώς και οι τρεις βασικές κατηγορίες της τμηματοποίησης σε συνδυασμό με ορισμένα παραδείγματα εφαρμογών. Παρουσιάζονται τα τεχνητά νευρωνικά δίκτυα (ANNs) και οι διάφορες κατηγορίες τους, όπως τα νευρωνικά δίκτυα εμπρόσθιας τροφοδότησης (FFNs), τα συνελικτικά νευρωνικά δίκτυα (CNNs) και τα επαναλαμβανόμενα νευρωνικά δίκτυα (RNNs), τα οποία χρησιμοποιούνται ευρέως σε εργασίες σημασιολογικής τμηματοποίησης. Το επίκεντρο του ενδιαφέροντος όμως βρίσκεται στα τρια μοντέλα UNet, DeepLab και PSPNet, για τα οποία πραγματοποιείται λεπτομερής ανάλυση της αρχιτεκτονικής τους και αναφέρονται ορισμένες από τις υπερπαράμετρους εκπαίδευσης, και τις μεθόδους αξιολόγησης τους. Τέλος, περιγράφεται η μεθοδολογία της συγκεκριμένης έρευνας και παρατίθενται τα αποτελέσματα που εξήχθησαν.

Η συγκεκριμένη προσπάθεια ανίχνευσης των τριών μορφολογικών χαρακτηριστικών (ICM, TE και ZP) των βλαστοκύστεων οδήγησε στην ανάδειξη του μοντέλου UNet ως καταλληλότερου, καθώς παρουσιάζει συνολικά την καλύτερη απόδοση, σε σύγκριση με τα άλλα δύο μοντέλα. Πρόκειται για ένα σταθερό μοντέλο, με καλή γενίκευση και χωρίς προβλήματα υπερπροσαρμογής. Αναφέρεται μια τιμή της μετρικής αξιολόγησης IoU του 67.05% για την αναγνώριση της ICM, μια τιμή του 56.88% για την αναγνώριση της TE και μια τιμή του 63.65% για την αναγνώριση της ZP. Τέλος, παρατίθενται ορισμένες προτάσεις και βελτιώσεις, οι οποίες πιθανόν θα οδηγήσουν στην επίτευξη καλύτερων αποτελεσμάτων τόσο για τα μοντέλα DeepLab και PSPNet, όσο και σε μελλοντικές έρευνες στο τομέα της σημασιολογικής τμηματοποίησης ιατρικών εικόνων.

Λέξεις – Κλειδιά: ανάλυση βλαστοκύστεων, εξωσωματική γονιμοποίηση (IVF), εσωτερική κυτταρική μάζα (ICM), τροφεκτοδερμία (TE), πυκνή μάζα (ZP), τμηματοποίηση, σημασιολογική τμηματοποίηση, UNet, DeepLab, PSPNet.

Abstract

The aspiration of this thesis is the development and comparison of three semantic segmentation models, UNet, DeepLab, and PSPNet, for the detection of three morphological characteristics (ICM, TE, and ZP) from blastocyst images, aiming to assess embryo suitability for in vitro fertilization (IVF). These models exhibit different behaviors with respect to the dataset used for training and evaluation.

This thesis highlights the significance of image segmentation in machine learning, particularly in the medical field, as well as the three main categories of segmentation, along with relevant application examples. It explores artificial neural networks (ANNs) and their various types, such as feed-forward neural networks (FFNs), convolutional neural networks (CNNs), and recurrent neural networks (RNNs), which are widely used in semantic segmentation tasks. However, the primary focus is on the UNet, DeepLab, and PSPNet models, providing a detailed analysis of their architectures, the hyperparameters used for training, and their evaluation methods. Finally, the research methodology is outlined, followed by a presentation of the obtained results.

This study's effort to detect the three morphological characteristics (ICM, TE, and ZP) in blastocysts identified the UNet model as the most suitable, as it demonstrated the best overall performance compared to the other two models. The UNet model proved to be the most stable, with strong generalization capabilities and no overfitting issues. The Intersection over Union (IoU) metric evaluation for the detection of ICM is reported to be 67.05%, 56.88% for TE, and 63.65% for ZP. Lastly, suggestions for improvements are provided in order to enhance the performance of the DeepLab and PSPNet models, as well as strengthen the foundation for the future research in the field of semantic segmentation of medical images, particularly blastocyst images.

Keywords: blastocyst analysis, in vitro fertilization (IVF), inner cell mass (ICM), trophectoderm (TE), zona pellucida (ZP), segmentation, semantic segmentation, UNet, DeepLab, PSPNet.

1. Εισαγωγή

1.1 Προσδιορισμός Προβλήματος

Η εξωσωματική γονιμοποίηση είναι μια διαδικασία κατά την οποία οι ωοθήκες της γυναίκας διεγείρονται μέσω φαρμάκων γονιμότητας, με σκοπό να παραχθούν περισσότερα από ένα ωάρια τον μήνα. Τα γονιμοποιημένα ωάρια καλλιεργούνται και αξιολογούνται σε ειδικούς χώρους με ελεγχόμενες περιβαλλοντικές συνθήκες. Αρχικά, τα έμβρυα φτάνουν στο στάδιο της βλαστοκύστης και αμέσως μετά επιλέγονται για εμφύτευση [6]. Υπάρχουν αρκετές μέθοδοι για την αξιολόγηση της ποιότητας των εμβρύων. Την παρούσα χρονική στιγμή, η πιο αξιόπιστη μέθοδος είναι ο προεμφυτευτικός γενετικός έλεγχος (PGS), ο οποίος έχει υψηλή ακρίβεια πρόβλεψης των μη εμφυτευτικών εμβρύων αλλά μέτρια θετική προγνωστική αξία [1][2]. Παρόλα αυτά, αν και αξιόπιστη, η μέθοδος αυτή δεν χρησιμοποιείται αρκετά εξαιτίας του υψηλού κόστους. Η ευρέως χρησιμοποιούμενη μέθοδος για την αξιολόγηση των εμβρύων βασίζεται στα μορφολογικά χαρακτηριστικά. Ωστόσο, αυτή η μέθοδος δεν είναι πάντα τόσο ακριβής, καθώς η αξιολόγηση εξαρτάται αποκλειστικά από τον εκάστοτε ειδικό. Η έλλειψη αυτοματοποιημένων μεθόδων αποτελεί ένα μεγάλο ζήτημα, καθώς η χρήση τους θα βοηθούσε στην λεπτομερέστερη ανάλυση και αξιολόγηση μορφολογικών χαρακτηριστικών, όπως οι περιοχές της εσωτερικής κυτταρικής μάζας (ICM), της τροφεκτοδερμίας (TE) και της πυκνής ζώνης (ZP) [6]. Το ερευνητικό πρόβλημα που θα μας απασχολήσει στην παρούσα Διπλωματική εργασία αφορά την ανίχνευση αυτών των περιοχών με την χρήση διάφορων μοντέλων σημασιολογικής τμηματοποίησης. Η επίλυση του παραπάνω προβλήματος θα οδηγήσει τους εμβρυολόγους στην λήψη ακριβέστερων αποφάσεων σε θέματα όπως ο προσδιορισμός των εμβρύων ως κατάλληλα ή όχι για εξωσωματική γονιμοποίηση.

1.2 Στόχος Διπλωματικής Εργασίας

Στόχος της Διπλωματικής εργασίας είναι η μελέτη και η ανάπτυξη τριών μοντέλων σημασιολογικής τμηματοποίησης, μέσω των οποίων θα επιτυγχάνεται η ανίχνευση των τριών χαρακτηριστικών (ICM, TE και ZP) των βλαστοκύστεων. Πιο αναλυτικά, πρόκειται για την υλοποίηση και την εκπαίδευση των τριών μοντέλων UNet, DeepLab και PSPNet, μέσω των οποίων θα πραγματοποιείται η τμηματοποίηση εικόνων βλαστοκύστεων και η

εξαγωγή τριών προβλεπόμενων μασκών, μια για κάθε χαρακτηριστικό τους (ICM, TE και ZP), με στόχο την επίτευξη υψηλών επιδόσεων, δηλαδή υψηλών τιμών για την μετρική αξιολόγησης IoU.

1.3 Συνοπτική Παρουσίαση Κεφαλαίων

Στο Κεφάλαιο 2, πραγματοποιείται τόσο ο προσδιορισμός της έννοιας της τμηματοποίησης, όσο και η ανάλυσή των τριών βασικών κατηγοριών της. Επιπλέον, παρατίθενται ορισμένα παραδείγματα διάφορων εφαρμογών, αναδεικνύοντας την σημαντικότητα της τμηματοποίησης σε ένα μεγάλο φάσμα επιστημονικών κλάδων. Τέλος, παρουσιάζεται μια κριτική επισκόπηση του βασικού βιβλιογραφικού υποβάθρου του θέματος της παρούσας διπλωματικής εργασίας, με σκοπό την ανάδειξη κενών, για τα οποία χρειάζεται έρευνα.

Στο Κεφάλαιο 3, πραγματοποιείται ο προσδιορισμός της έννοιας της βαθιάς μάθησης και μια λεπτομερής ανάλυση τόσο των τεχνητών νευρωνικών δικτύων (ANNs), όσο και των διάφορων κατηγοριών τους, όπως τα νευρωνικά δίκτυα εμπρόσθιας τροφοδότησης (FFNs), τα συνελικτικά νευρωνικά δίκτυα (CNNs) και τα επαναλαμβανόμενα νευρωνικά δίκτυα (RNNs). Επιπλέον, περιγράφονται οι αρχιτεκτονικές συγκεκριμένων μοντέλων, όπως του UNet, του DeepLab και του PSPNet, καθώς επίσης και οι υπερπαράμετροι εκπαίδευσης και οι μέθοδοι αξιολόγησης τους.

Στο Κεφάλαιο 4, παρουσιάζεται η μεθοδολογία που ακολουθήθηκε, με σκοπό την επίτευξη του στόχου της εργασίας. Πιο αναλυτικά, αναφέρεται το πως πραγματοποιήθηκε η συλλογή και η προετοιμασία των δεδομένων πριν την εισαγωγή τους στα μοντέλα, πως υλοποιήθηκε κάθε μοντέλο ξεχωριστά και ποιες υπερπαράμετροι εκπαίδευσης χρησιμοποιήθηκαν για τα εν λόγω μοντέλα. Τέλος, αναφέρεται το πως πραγματοποιήθηκε η διαδικασία της εκπαίδευσης και των προβλέψεων των μοντέλων, καθώς και ποιες μετρικές αξιολόγησης χρησιμοποιήθηκαν.

Στο Κεφάλαιο 5, παρουσιάζονται τα αποτελέσματα που εξήχθησαν από την συγκεκριμένη έρευνα, με σκοπό την επίτευξη του στόχου της εργασίας. Τα αποτελέσματα αφορούν την επιρροή των υπερπαραμέτρων εκπαίδευσης (αριθμός εποχών, μέγεθος παρτίδας και ρυθμός εκμάθησης) στην απόδοση των μοντέλων. Επιπλέον, με βάση τα αποτελέσματα, αναδεικνύεται ποιο μοντέλο έχει την καλύτερη

επίδοση, δηλαδή ποιο μοντέλο παρουσιάζει τις μεγαλύτερες τιμές για την μετρική αξιολόγησης IoU.

Στο Κεφάλαιο 6, αναφέρονται όλα τα συμπεράσματα που εξήχθησαν με την εκπλήρωση της Διπλωματικής εργασίας, καθώς και μελλοντικές βελτιώσεις που πιθανόν να οδηγήσουν σε υψηλότερες τιμές για την μετρική αξιολόγησης IoU.

2. Τμηματοποίηση Εικόνας (Image Segmentation)

2.1 Εισαγωγή

Ο όρος τμηματοποίηση ή κατάτμηση εικόνας αναφέρεται στην διαδικασία διαίρεσης μιας εικόνας σε πολλαπλές περιοχές ή τμήματα. Κάθε προκύπτουσα περιοχή ή τμήμα αντιστοιχεί σε ένα αντικείμενο ή χαρακτηριστικό της εικόνας. Η τμηματοποίηση είναι ένας κλάδος της ψηφιακής επεξεργασίας εικόνας που στοχεύει στην απλοποίηση των χαρακτηριστικών της εικόνας, ώστε να είναι πιο εύκολη η ανάλυσή τους [3][4]. Το αποτέλεσμα της διαδικασίας της κατάτμησης είναι μια εικόνα χωρισμένη σε ξεχωριστά τμήματα, τα οποία εμφανίζουν κάποια κοινά χαρακτηριστικά.

Έχουν αναπτυχθεί αρκετές τεχνικές τμηματοποίησης εικόνας. Ορισμένες από αυτές τις τεχνικές είναι η κατωφλίωση (thresholding), η ανίχνευση ακμών (edge detection), τα ενεργά περιγράμματα (active contours), η τμηματοποίηση βάσει περιοχής (region-based segmentation), η τμηματοποίηση με βάση την γραφική αναπαράσταση και η μορφολογική επεξεργασία. Η χρήση τέτοιου είδους τεχνικών κατάτμησης βοηθάει στο διαχωρισμό και την ομαδοποίηση των εικονοστοιχείων μιας εικόνας, την αντιστοίχισή τους σε επιθυμητές κατηγορίες και στην συνέχεια την ταξινόμηση των εικονοστοιχείων αυτών σύμφωνα με αυτές τις κατηγορίες.

Η κατάτμηση εικόνας χρησιμοποιείται σε εργασίες που αφορούν την ανίχνευση αντικειμένων ή ορίων σε εικόνες. Ορισμένες εφαρμογές της τμηματοποίησης εικόνας είναι η ανάκτηση εικόνας βάσει περιεχομένου, η ανάλυση ιατρικών εικόνων, η ανίχνευση αντικειμένων, η αναγνώριση προσώπου, τα συστήματα ελέγχου κυκλοφορίας κ.α. Υπάρχουν τρεις κατηγορίες τμηματοποίησης, οι οποίες είναι:

1) Σημασιολογική Τμηματοποίηση (Semantic Segmentation)
2) Τμηματοποίηση Παρουσίας (Instance Segmentation)
3) Πανοπτική Τμηματοποίηση (Panoptic Segmentation)

Στην συνέχεια ακολουθεί μια λεπτομερέστερη περιγραφή αυτών των τριών κατηγοριών τμηματοποίησης.

2.1.1 Σημασιολογική Τμηματοποίηση (Semantic Segmentation)

Η σημασιολογική τμηματοποίηση είναι μια εργασία υπολογιστικής όρασης, κατά την οποία επιτυγχάνεται η διάκριση διαφορετικών κατηγοριών αντικειμένων σε μια εικόνα. Πιο αναλυτικά, ο όρος σημασιολογική τμηματοποίηση αναφέρεται στην ανάθεση μιας ετικέτας σε κάθε εικονοστοιχείο της εικόνας. Η εν λόγω ετικέτα αντιστοιχεί σε ένα αντικείμενο ή στην περιοχή στην οποία ανήκει το εικονοστοιχείο [5]. Η εφαρμογή μιας τέτοιου είδους τμηματοποίησης οδηγεί στην δημιουργία ενός χάρτη τμηματοποίησης από την αρχική εικόνα. Ουσιαστικά, πρόκειται για την ανακατασκευή της αρχικής εικόνας, όπου σε κάθε εικονοστοιχείο έχει αποδοθεί ένα χρώμα, ανάλογα με την κλάση στην οποία ανήκει, με σκοπό την γέννηση μιας μάσκας. Η μάσκα αυτή, που συχνά αναφέρεται ως μάσκα κατάτμησης, είναι μια περιοχή της εικόνας, η οποία έχει διαφοροποιηθεί από τις υπόλοιπες περιοχές. Ένα απλό παράδειγμα εφαρμογής της σημασιολογικής τμηματοποίησης παρουσιάζεται στην Εικόνα 2.1. Ο χάρτης τμηματοποίησης στο δεξιά μέρος της Εικόνας 2.1, περιέχει πέντε μάσκες τμηματοποίησης: μια για τους ανθρώπους, μια για τον χαρταετό, μια για τον ουρανό, μια για την θάλασσα και μια για την αμμουδιά.

Εικόνα 2.1 – Εφαρμογή Σημασιολογικής Τμηματοποίησης

Είναι άξιο να σημειωθεί ότι, αν υπάρξουν δύο αντικείμενα της ίδιας κατηγορίας στην αρχική εικόνα, τότε δεν διακρίνονται ως ξεχωριστά αντικείμενα στον χάρτη τμηματοποίησης. Για την περίπτωση της διάκρισης ξεχωριστών αντικειμένων, τα οποία ανήκουν στην ίδια κατηγορία, υπάρχει μια άλλη κατηγορία μοντέλων, τα οποία είναι γνωστά ως μοντέλα τμηματοποίησης παρουσίας.

Η επίτευξη υψηλής ακρίβειας στην σημασιολογική τμηματοποίηση φαντάζει μια εξαιρετικά δύσκολη υπόθεση, καθώς υπάρχει μεγάλη ποικιλομορφία και αρκετές αλληλοκαλύψεις μεταξύ των αντικειμένων της εικόνας. Για την επίλυση του

συγκεκριμένου ζητήματος, τα μοντέλα σημασιολογικής τμηματοποίησης χρησιμοποιούν πολύπλοκα νευρωνικά δίκτυα. Τα συνελικτικά νευρωνικά δίκτυα (CNN), τα οποία βοηθούν στην εξαγωγή χαρακτηριστικών από την αρχική εικόνα, αποτελούνται από ένα πλήθος συνελικτικών στρωμάτων και στην έξοδό τους παράγεται ο χάρτης τμηματοποίησης.

2.1.2 Τμηματοποίηση Παρουσίας (Instance Segmentation)

Στην υποενότητα 2.1.1 αναφέρθηκε ότι ο στόχος της σημασιολογικής τμηματοποίησης είναι η πρόβλεψη της κατηγορίας για κάθε εικονοστοιχείο μιας εικόνας, το οποίο οδηγεί στην γέννηση μιας μάσκας για κάθε κατηγορία. Η τμηματοποίηση παρουσίας είναι ένας συνδυασμός της σημασιολογικής τμηματοποίησης και της ανίχνευσης αντικειμένων (Object Detection). Ο όρος ανίχνευση αντικειμένων αναφέρεται στην ανίχνευση διαφορετικών αντικειμένων σε μια εικόνα. Επομένως, κατά την τμηματοποίηση παρουσίας, κάθε εικονοστοιχείο της εικόνας κατατάσσεται σε μια κατηγορία και ταυτόχρονα πραγματοποιείται ανίχνευση αντικειμένων. Κάθε εξαγόμενη μάσκα δεν θα περιέχει όλα τα αντικείμενα μιας συγκεκριμένης κατηγορίας, όπως συνέβαινε στην σημασιολογική τμηματοποίηση, αλλά μια μόνο εμφάνιση της κατηγορίας. Ένα αντικείμενο μιας συγκεκριμένης κατηγορίας θα έχει διαφορετική μάσκα από ένα αντικείμενο της ίδιας κατηγορίας, όπως παρουσιάζεται στην Εικόνα 2.2. Τέλος, αξίζει να σημειωθεί πως η τμηματοποίηση παρουσίας αποτελεί μια πολύπλοκη διαδικασία, καθώς το μοντέλο πρέπει να διακρίνει με μεγάλη ακρίβεια μεταξύ διαφορετικών περιπτώσεων αντικειμένων της ίδιας κατηγορίας. Επομένως, το μοντέλο θα πρέπει να είναι σε θέση να ξεπεράσει τυχόν αλληλοκαλύψεις αντικειμένων, αλλά και τα διαφορετικά σχήματα και μεγέθη που αυτά διαθέτουν.

Εικόνα 2.2 – Εφαρμογή Τμηματοποίησης Παρουσίας

2.1.3 Πανοπτική Τμηματοποίηση (Panoptic Segmentation)

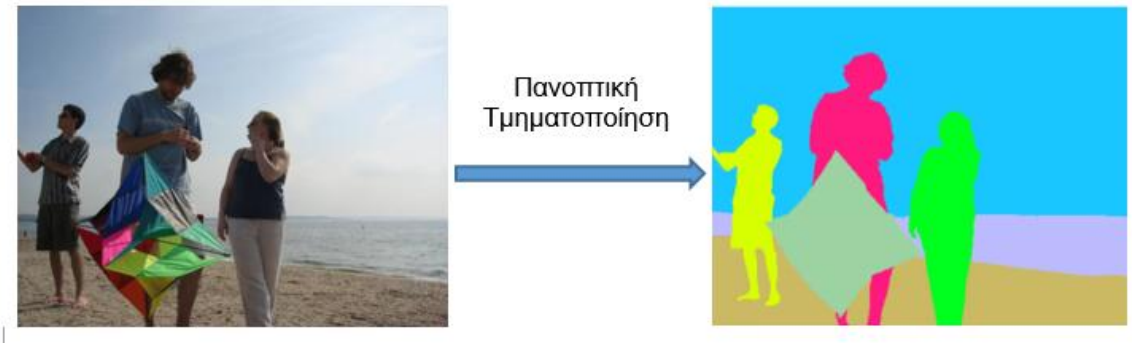
Η πανοπτική τμηματοποίηση είναι ακόμη μια εργασία υπολογιστικής όρασης, η οποία προκύπτει από τον συνδυασμό της σημασιολογικής τμηματοποίησης και της τμηματοποίησης παρουσίας. Όπως και στην σημασιολογική τμηματοποίηση, σε κάθε εικονοστοιχείο εκχωρείται μια κλάση, αλλά στην περίπτωση που υπάρχουν πολλές παρουσίες μιας κλάσης, τότε είναι γνωστό ποιο εικονοστοιχείο ανήκει σε ποια παρουσία της κλάσης. Με αυτόν τον τρόπο, γίνεται ο συνδυασμός των δύο προαναφερθέντων ειδών τμηματοποίησης και προκύπτει η πανοπτική τμηματοποίηση.

Κάθε εικονοστοιχείο σε μια εικόνα έχει διαφορετικό χρώμα. Για παράδειγμα, στην Εικόνα 2.3, ο ουρανός έχει γαλάζιο χρώμα, η θάλασσα έχει μωβ, η αμμουδιά έχει καφέ, ο χαρταετός έχει πράσινο και οι άνθρωποι είναι χρωματισμένοι σε διαφορετικές αποχρώσεις π.χ. κίτρινο, κόκκινο και ανοιχτό πράσινο. Επομένως, η πανοπτική τμηματοποίηση βοηθά στον προσδιορισμό κάθε αντικειμένου σε μια εικόνα και τα διακρίνει μεταξύ τους, παρέχοντας ένα μοναδικό χρώμα μάσκας για το καθένα.

Με την εφαρμογή πανοπτικής τμηματοποίησης σε μια εικόνα, κάθε εικονοστοιχείο της εικόνας διαθέτει δύο τιμές. Η πρώτη τιμή υποδεικνύει την κατηγορία του εικονοστοιχείου, ενώ η δεύτερη τιμή αποτελεί έναν αριθμό παρουσίας. Εικονοστοιχεία που βρίσκονται σε περιοχές, οι οποίες είναι δύσκολο να ποσοτικοποιηθούν (π.χ. ουρανός, θάλασσα, αμμουδιά), μπορεί να διαθέτουν έναν ή και κανέναν αριθμό παρουσίας. Από την άλλη πλευρά, εικονοστοιχεία που ανήκουν σε μετρήσιμα αντικείμενα (π.χ. άνθρωπος), διαθέτουν έναν και μοναδικό αριθμό παρουσίας.

Ο στόχος της πανοπτικής τμηματοποίησης είναι η ακριβής διάκριση μεταξύ διαφορετικών αντικειμένων της ίδιας κατηγορίας. Τα μοντέλα που χρησιμοποιούνται για την εκτέλεση πανοπτικής τμηματοποίησης ακολουθούν μια συγκεκριμένη διαδικασία. Αρχικά, πραγματοποιείται η ανίχνευση των αντικειμένων και η ταξινόμηση διαφορετικών περιπτώσεων αυτών. Στην συνέχεια, πραγματοποιείται η τμηματοποίηση με σκοπό την δημιουργία μασκών για κάθε περίπτωση αντικειμένου.

Η συγκεκριμένη τεχνική τμηματοποίησης έχει εφαρμογές σε τομείς όπως η ιατρική, η ρομποτική και η αυτόνομη οδήγηση. Στην αυτόνομη οδήγηση χρησιμοποιείται για την ανίχνευση οχημάτων και πεζών, με σκοπό την λήψη ασφαλέστερων αποφάσεων από τα αυτόνομα οχήματα. Στην ρομποτική, χρησιμοποιείται για την ανίχνευση αντικειμένων, με σκοπό την καλύτερη αλληλεπίδραση του ρομπότ με το περιβάλλον του.



Εικόνα 2.3 – Εφαρμογή Πανοπτικής Τμηματοποίησης

2.2 Παραδείγματα Εφαρμογών Σημασιολογικής Τμηματοποίησης

Η σημασιολογική τμηματοποίηση αποτελεί μια ισχυρή τεχνική της υπολογιστικής όρασης. Καλύπτει ένα μεγάλο εύρος εφαρμογών σε διαφορετικούς επιστημονικούς κλάδους όπως η Ιατρική, η Ρομποτική, η Γεωργία κ.α. Ακολουθεί η παρουσίαση ορισμένων εφαρμογών σημασιολογικής τμηματοποίησης.

2.2.1 Ανάλυση Ιατρικών Εικόνων

Η σημασιολογική τμηματοποίηση έχει παίξει σημαντικό ρόλο στον τομέα της Ιατρικής, καθώς βοηθάει στην ανίχνευση ανωμαλιών ή ασθενειών, μέσα από την τμηματοποίηση ιατρικών εικόνων, οι οποίες προκύπτουν από μαγνητικούς τομογράφους ή ακτινογραφικά μηχανήματα. Πιο συγκεκριμένα, επιτυγχάνεται ακριβής ανίχνευση όγκων, νευρολογικών διαταραχών, καρδιαγγειακών παθήσεων, αλλά και η χαρτογράφηση ιστών και οργάνων. Με αυτόν τον τρόπο, έχει βελτιωθεί ραγδαία η διαδικασία της διάγνωσης, το οποίο βοηθάει αρκετά στην λήψη τεκμηριωμένων αποφάσεων σχετικά με τα σχέδια θεραπείας.

Μια εξαιρετικά σημαντική εφαρμογή της σημασιολογικής τμηματοποίησης είναι η τμηματοποίηση του όγκου. Προσφέρει πολύτιμες πληροφορίες για την θέση, το μέγεθος και τα χαρακτηριστικά των όγκων. Βοηθάει τους ακτινολόγους να πετύχουν τον ακριβή εντοπισμό και την οριοθέτηση του όγκου, το οποίο αποτελεί μια απαραίτητη πληροφορία για αποτελεσματικές στρατηγικές θεραπείας (π.χ. ακτινοθεραπεία), τον προγραμματισμό χειρουργικών επεμβάσεων και την μελλοντική παρακολούθηση του όγκου.

Τέλος, η σημασιολογική τμηματοποίηση, εκτός από την ανίχνευση ασθενειών, βοηθάει και σε αρκετές ιατρικές έρευνες που αφορούν την αποτελεσματικότητα νέων φαρμάκων. Οι επιστήμονες του εν λόγω επιστημονικού κλάδου αναλύουν τεράστια σύνολα δεδομένων, τα οποία αποτελούνται από ιατρικές εικόνες, και εξάγουν συμπεράσματα σχετικά με την εξέλιξη της ασθένειας, την ανταπόκριση στην θεραπεία και την αποτελεσματικότητα των φαρμάκων.

2.2.2 Αυτόνομη Οδήγηση

Στις μέρες μας, η σημασιολογική τμηματοποίηση έχει συμβάλει καθοριστικά στην αυτονομία των σύγχρονων οχημάτων. Η πρόκληση στην αυτόνομη οδήγηση είναι ο ακριβής εντοπισμός και η κατανόηση του οδικού περιβάλλοντος. Με την τμηματοποίηση των εικονοστοιχείων μιας εικόνας σε διαφορετικές κατηγορίες (π.χ οχήματα, πεζοί, δέντρα, κολώνες), τα αυτόνομα οχήματα αντιλαμβάνονται τα διαφορετικά αντικείμενα και τα χαρακτηριστικά του δρόμου και είναι σε θέση να πλοηγούνται στον δρόμο με ασφάλεια. Επιπλέον, βάσει της εν λόγω τμηματοποίησης, τα οχήματα μπορούν να ανιχνεύουν τα σήματα, τα όρια των λωρίδων κυκλοφορίας και να λαμβάνουν τεκμηριωμένες αποφάσεις σε πραγματικό χρόνο.

2.2.3 Ρομποτική Όραση

Με την συνεχόμενη τεχνολογική εξέλιξη, τα ρομπότ έχουν εισχωρήσει στην καθημερινότητα μας, αλλά και σε αρκετούς επιστημονικούς κλάδους. Η πρόκληση στην ρομποτική όραση είναι η εκτέλεση απαιτητικών οπτικών εργασιών από τα ρομπότ, με σκοπό την πλοήγηση σε πολύπλοκα περιβάλλοντα. Τα ρομπότ πρέπει να βρίσκονται σε θέση να αντιλαμβάνονται την διαφορά μεταξύ των αντικειμένων και να αποφεύγουν αποτελεσματικά ενδεχόμενες συγκρούσεις. Η σημασιολογική τμηματοποίηση παρέχει υψηλή ακρίβεια σχετικά με την ικανότητα των ρομπότ να βλέπουν και να κατανοούν το περιβάλλον τους. Με αυτόν τον τρόπο, τα ρομπότ είναι ικανά να διαχωρίζουν τις διάφορες περιοχές και αντικείμενα του περιβάλλοντός τους σε κατηγορίες (π.χ. καρέκλα, κτίριο, πεζοδρόμιο, δρόμος), με σκοπό την αυτόνομη πλοήγησή τους.

2.2.4 Ανάλυση Εικόνων Καλλιεργειών

Για αρκετά χρόνια, οι αγρότες δεν ήταν σε θέση να ανιχνεύσουν τυχόν παράσιτα ή διάφορες ασθένειες, ώστε να βοηθήσουν τις καλλιέργειές τους. Την λύση στο συγκεκριμένο πρόβλημα ήρθε να δώσει η σημασιολογική τμηματοποίηση, η οποία έχει εισέλθει και στον κλάδο της Γεωργίας. Οι αγρότες πλέον έχουν την δυνατότητα να αναλύουν εικόνες από τις καλλιέργειές τους, που προέρχονται είτε από κάμερες είτε από drones, με σκοπό την ακριβή ανίχνευση ασθενειών ή τον εντοπισμό τυχόν παρασίτων. Με αυτόν τον τρόπο, οι αγρότες μπορούν να διακρίνουν ποια τμήματα από τις καλλιέργειές τους είναι υγιή και ποια τμήματα έχουν μολυνθεί. Επιπλέον, η εφαρμογή των φυτοφαρμάκων μπορεί να πραγματοποιηθεί στοχευμένα στο τμήμα του χωραφιού που έχει μολυνθεί, το οποίο οδηγεί σε μια αλυσίδα θετικών αποτελεσμάτων, όπως η μείωση της χρήσης φυτοφαρμάκων, η μείωση των αρνητικών επιδράσεων στο έδαφος, η μείωση των περιβαλλοντικών συνθηκών και η μείωση του κόστους.

2.2.5 Τηλεπισκόπηση και Γεωχωρική Ανάλυση

Η σημασιολογική τμηματοποίηση έχει ευρεία χρήση στον τομέα της τηλεπισκόπησης και της γεωχωρικής ανάλυσης. Μια από τις βασικές εφαρμογές αποτελεί η ακριβή και η λεπτομερή χαρτογράφηση της γης. Η εικόνα διαχωρίζεται σε διαφορετικές κατηγορίες (π.χ. δάση, ωκεανοί, αστικές περιοχές) και πραγματοποιείται η ταξινόμηση των περιοχών της γης. Οι πληροφορίες αυτές είναι άξιες σημασίας για τον πολεοδομικό σχεδιασμό, την αξιολόγηση των αλλαγών με το πέρασμα των χρόνων κ.α. Επιπλέον, είναι εφικτή η παρακολούθηση της βλάστησης μέσα από την τμηματοποίηση εικόνων, οι οποίες προέρχονται από δορυφόρους. Με τον τρόπο αυτό, μπορεί να αναλυθεί η υγεία και η κατανομή της βλάστησης σε μια συγκεκριμένη περιοχή, το οποίο είναι σημαντικό για οικολογικές μελέτες και την αξιολόγηση των επιπτώσεων της κλιματικής αλλαγής στα φυτά.

Η σημασιολογική κατάτμηση βοηθάει σημαντικά στην ανάλυση και την εξαγωγή συμπερασμάτων για γεωγραφικά δεδομένα μεγάλης κλίμακας. Παρέχει ακριβή και λεπτομερή τμηματοποίηση εικόνων, οδηγεί στην λήψη τεκμηριωμένων αποφάσεων με αποτέλεσμα την διασφάλιση της βιωσιμότητας στην γη, την προστασία του περιβάλλοντος και την βελτιστοποίηση των πόρων.

2.3 Βιβλιογραφική Ανασκόπηση

Με την πάροδο του χρόνου διακεκριμένοι επιστήμονες ανέπτυξαν αρκετές ερευνητικές μελέτες, μέσω των οποίων αποδεικνύεται ότι η εφαρμογή συνελικτικών νευρωνικών δικτύων (CNN) αποτελεί την καταλληλότερη επιλογή στο τομέα της τμηματοποίησης ιατρικών εικόνων. Ωστόσο οι έρευνες δεν επικεντρώθηκαν μόνο στην τμηματοποίηση ιατρικών δεδομένων, όπως στην ανάλυση βλαστοκύστεων [6], αλλά και στην εξαγωγή οδικών δικτύων [7] και στην ανίχνευση παθολογιών [8] κ.α. Το 2015, ο Olaf Ronneberger, ο Philipp Fischer και ο Thomas Brox δημιούργησαν την αρχιτεκτονική του δικτύου UNet [9], το οποίο αποτελεί μια βελτιστοποιημένη εκδοχή των πλήρως συνελικτικών δικτύων (FCN) [10]. Τόσο το δίκτυο UNet όσο και διάφορες παραλλαγές του έπαιξαν σημαντικό ρόλο στην ανάπτυξη πιο σύγχρονων τεχνικών. Μια από αυτές τις παραλλαγές ήταν το VNet, το οποίο αναδείχθηκε από τους Fausto Milletarì, Nassir Navab και Seyed-Ahmad Ahmadi το 2016 [11]. Χαρακτηριστικό αυτών των δικτύων είναι ότι τα χαρακτηριστικά χαμηλού και υψηλού επιπέδου συνδυάζονται μέσω skip connections.

Ακολούθησε πληθώρα ερευνητικών μελετών, οι οποίες είχαν σκοπό την επίτευξη μεγαλύτερης ακρίβειας στην τμηματοποίηση και την πιο εύκολη προσαρμογή σε διαφορετικούς τύπους δεδομένων. Σε κάποιες από αυτές τις μελέτες πραγματοποιήθηκε η ενσωμάτωση μηχανισμού προσοχής στο δίκτυο UNet, το οποίο καθιερώθηκε με την ονομασία Attention UNet και βοήθησε αρκετά όσον αφορά την επίτευξη μεγαλύτερης ακρίβειας στην τμηματοποίηση ιατρικών εικόνων [12]. Σε μια άλλη μελέτη πραγματοποιήθηκε ο συνδυασμός υπολειμματικών συνδέσεων σε ένα επαναληπτικό δίκτυο, το οποίο είχε ως αποτέλεσμα καλύτερες αναπαραστάσεις χαρακτηριστικών [13]. Μια ακόμη αρχιτεκτονική που ενίσχυσε την ακρίβεια στην ανάλυση ιατρικών εικόνων είναι το ResUNet++. Στην εν λόγω αρχιτεκτονική έχουν ενσωματωθεί τεχνικές πολυκλίμακης ανάλυσης και πραγματοποιείται σύνθεση διαφορετικών χαρακτηριστικών [14].

Υπάρχουν και άλλα μοντέλα σημασιολογικής τμηματοποίησης, τα οποία έχουν βοηθήσει στην επίτευξη υψηλής ακρίβειας σε εργασίες τμηματοποίησης εικόνων. Χαρακτηριστικό παράδειγμα αποτελεί το DeepLab, το οποίο χρησιμοποιήθηκε από τον Chen και άλλους επιστήμονες το 2018 [19]. Το εν λόγω μοντέλο χρησιμοποιεί atrous convolutions καθώς και τυχαία πεδία (CRFs) για την ενσωμάτωση πληροφοριών πολλών επιπέδων. Μια νεότερη έκδοση, το DeepLabv3+, χρησιμοποιεί αρχιτεκτονικές encoder-decoder για την επίτευξη ακόμη μεγαλύτερης ακρίβειας. Ένα ακόμη μοντέλο

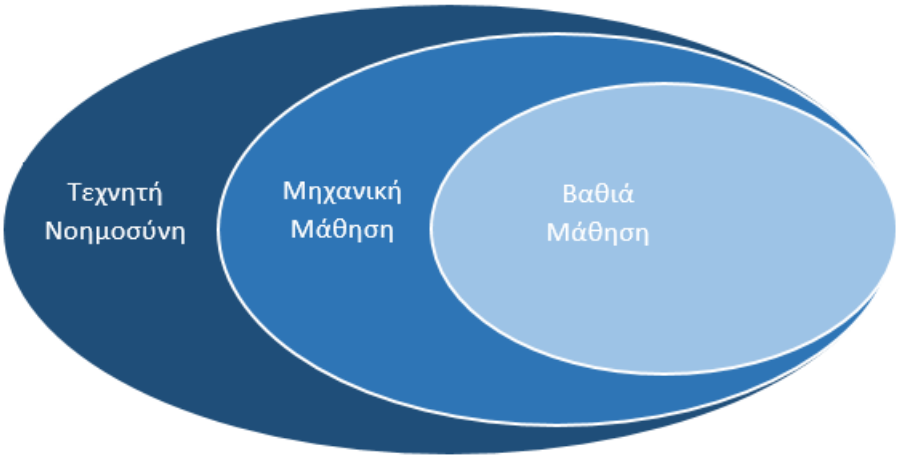
που αναδείχθηκε για την επίτευξη καλύτερων αποτελεσμάτων σε πολύπλοκα σύνολα δεδομένων είναι το PSPNet [16].

Η δημιουργία και η ανάπτυξη όλων αυτών των μοντέλων σημασιολογικής τμηματοποίησης έχει βοηθήσει σε μεγάλο βαθμό, όμως εμφανίζονται ακόμη αρκετές δυσκολίες. Η ανάγκη για μεγαλύτερο όγκο δεδομένων για εκπαίδευση αλλά και η ανάγκη για βελτίωση της ικανότητας των μοντέλων να γενικεύουν, είναι κάποιες από αυτές τις δυσκολίες που είναι απαραίτητο να αντιμετωπιστούν. Η παρούσα έρευνα έχει ως στόχο την σύγκριση και την ενσωμάτωση διάφορων τεχνικών για την βελτιστοποίηση της τμηματοποίησης ιατρικών εικόνων, με σκοπό την αντιμετώπιση των δυσκολιών που μόλις προαναφέρθηκαν.

3. Βαθιά Μάθηση και Τεχνητά Νευρωνικά Δίκτυα (Deep Learning and Artificial Neural Networks)

3.1 Εισαγωγή

Η βαθιά μάθηση γίνεται όλο και δημοφιλέστερη, εξαιτίας της ικανότητάς της να επιλύει πολύπλοκα προβλήματα, με την χρήση νευρωνικών δικτύων. Ορισμένες από τις εφαρμογές της είναι η επεξεργασία φυσικής γλώσσας, η αναγνώριση ομιλίας, η ανάλυση ιατρικών εικόνων κ.α. Η βαθιά μάθηση ανήκει στο ευρύτερο πεδίο της μηχανικής μάθησης, όπως παρουσιάζεται στην Εικόνα 3.1. Ο όρος βαθιά αναφέρεται στο πλήθος των επιπέδων που απαρτίζουν ένα νευρωνικό δίκτυο, τα οποία μπορεί να είναι από τρία έως μερικές εκατοντάδες ή και χιλιάδες. Ένα νευρωνικό δίκτυο, λόγω της βαθύτητάς του, είναι σε θέση να ανιχνεύει πιο περίπλοκες δομές, χωρίς να χρειάζονται μεγάλες ποσότητες δεδομένων.

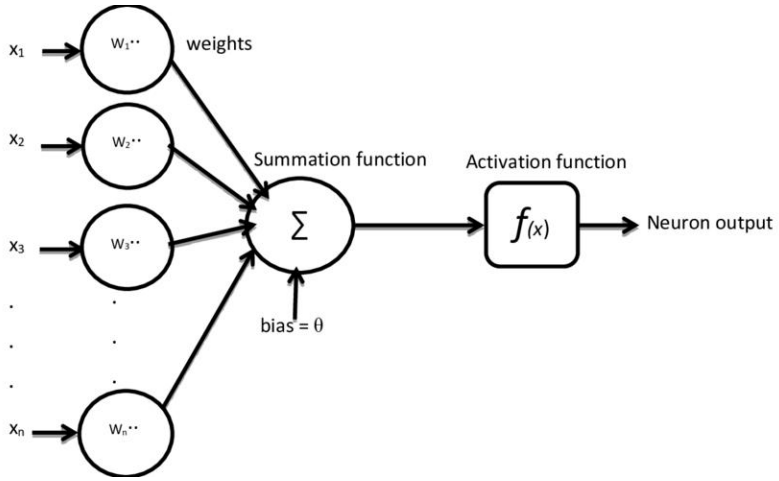


Εικόνα 3.1 – Τεχνητή Νοημοσύνη vs Μηχανική Μάθηση vs Βαθιά Μάθηση

Στις επόμενες ενότητες θα πραγματοποιηθεί αναφορά στα τεχνητά νευρωνικά δίκτυα και στις διάφορες κατηγορίες που διαθέτουν, στις αρχιτεκτονικές συγκεκριμένων μοντέλων, καθώς επίσης και στις υπερπαραμέτρους και τις μεθόδους αξιολόγησης των μοντέλων.

3.2 Τεχνητά Νευρωνικά Δίκτυα (Artificial Neural Networks)

Τα νευρωνικά δίκτυα, ή αλλιώς τεχνητά νευρωνικά δίκτυα (ANNs), αποτελούν ένα υπολογιστικό μοντέλο, το οποίο είναι εμπνευσμένο από τη δομή και τη λειτουργία των βιολογικών νευρωνικών συστημάτων [20]. Ένα νευρωνικό δίκτυο αποτελείται από πολλαπλά επίπεδα διασυνδεδεμένων κόμβων, ή αλλιώς τεχνητών νευρώνων. Κάθε επίπεδο δέχεται μια είδοσο, η οποία αποτελεί την έξοδο του προηγούμενου επιπέδου, και παράγει μια έξοδο, η οποία αποτελεί την είσοδο του επόμενου επιπέδου. Ένα νευρωνικό δίκτυο αποτελείται από τρία είδη επιπέδων, το επίπεδο εισόδου (input layer), το κρυφό επίπεδο (hidden layer) και το επίπεδο εξόδου (output layer). Στο επίπεδο εισόδου εισέρχονται όλα τα δεδομένα, τα οποία πρέπει να επεξεργαστούν από το δίκτυο. Το κρυφό επίπεδο αποτελεί το ενδιάμεσο επίπεδο μεταξύ του επιπέδου εισόδου και του επιπέδου εξόδου. Πρέπει να σημειωθεί ότι το νευρωνικό δίκτυο μπορεί να διαθέτει ένα ή περισσότερα κρυφά επίπεδα. Το επίπεδο εξόδου είναι υπεύθυνο για τα αποτελέσματα που θα παραχθούν από τους υπολογισμούς του δικτύου. Οι νευρώνες κάθε επιπέδου λαμβάνουν εισόδους, υπολογίζουν το αθροισμά των εισόδων, εφαρμόζουν μια συνάρτηση ενεργοποίησης πάνω στο άθροισμα και τέλος παράγουν μια έξοδο [21]. Οι νευρώνες συνδέονται μεταξύ τους και η σημαντικότητα της σύνδεσής τους χαρακτηρίζεται από ένα βάρος. Οι τιμές των βαρών των συνδέσεων αλλάζουν συνεχώς κατά την εκπαίδευση, με σκοπό την βελτίωση της απόδοσης [22]. Στην Εικόνα 3.2 παρουσιάζεται η δομή ενός τεχνητού νευρώνα.



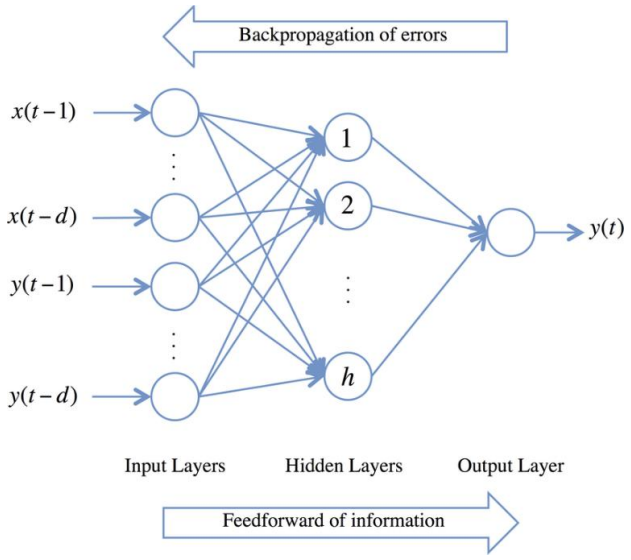
Εικόνα 3.2 – Δομή ενός τεχνητού νευρώνα [23]

Υπάρχουν διάφορες κατηγορίες νευρωνικών δικτύων. Ορισμένες από αυτές είναι:

I. Νευρωνικά Δίκτυα Εμπρόσθιας Τροφοδότησης (Feed-Forward Neural Networks)
II. Συνελικτικά Νευρωνικά Δίκτυα (Convolutional Neural Networks)
III. Επαναλαμβανόμενα Νευρωνικά Δίκτυα (Recurrent Neural Networks)

3.2.1 Νευρωνικά Δίκτυα Εμπρόσθιας Τροφοδότησης (Feed-Forward Neural Networks)

Ένα νευρωνικό δίκτυο εμπρόσθιας τροφοδότησης (FNN) αποτελείται από ένα επίπεδο εισόδου, ένα ή περισσότερα κρυφά επίπεδα και ένα επίπεδο εξόδου. Τα δεδομένα κατευθύνονται από την είσοδο προς την έξοδο, περνώντας διαδοχικά από όλους τους νευρώνες όλων των επιπέδων. Κάθε νευρώνας σε κάθε επίπεδο λαμβάνει εισόδους, υπολογίζει το αθροισμά των εισόδων, εφαρμόζει μια συνάρτηση ενεργοποίησης πάνω στο άθροισμα και τέλος παράγει μια έξοδο. Αυτή η διαδικασία ονομάζεται εμπρόσθια διάδοση. Για την εκπαίδευση των FNNs χρησιμοποιείται μια διαδικασία, η οποία ονομάζεται οπισθοδιάδοση σφάλματος (backpropagation). Η εν λόγω διαδικασία εκτελείται επαναληπτικά, ελαχιστοποιεί την συνάρτηση κόστους με σκοπό την προσαρμογή των βαρών και των biases. Στην Εικόνα 3.3 παρουσιάζεται ένα νευρωνικό δίκτυο εμπρόσθιας τροφοδότησης και η διαδικασία backpropagation.

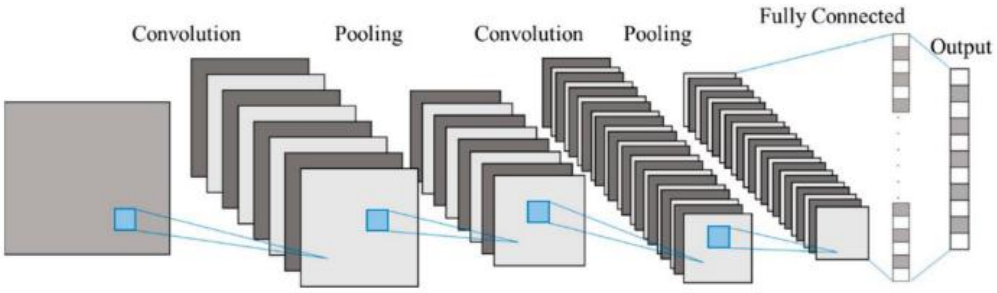


Εικόνα 3.3 – Απεικόνιση της αρχιτεκτονικής ενός FNN με backpropagation [24]

3.2.2 Συνελικτικά Νευρωνικά Δίκτυα (Convolutional Neural Networks)

Τα συνελικτικά νευρωνικά δίκτυα (CNNs) αποτελούν μια από τις πιο δημοφιλείς αρχιτεκτονικές νευρωνικών δικτύων, καθώς διακρίνεται για την επίτευξη υψηλών επιδόσεων σε προβλήματα υπολογιστικής όρασης [25]. Η αρχιτεκτονική ενός CNN παρουσιάζεται στην Εικόνα 3.4 και περιλαμβάνει τρία διαφορετικά επίπεδα:

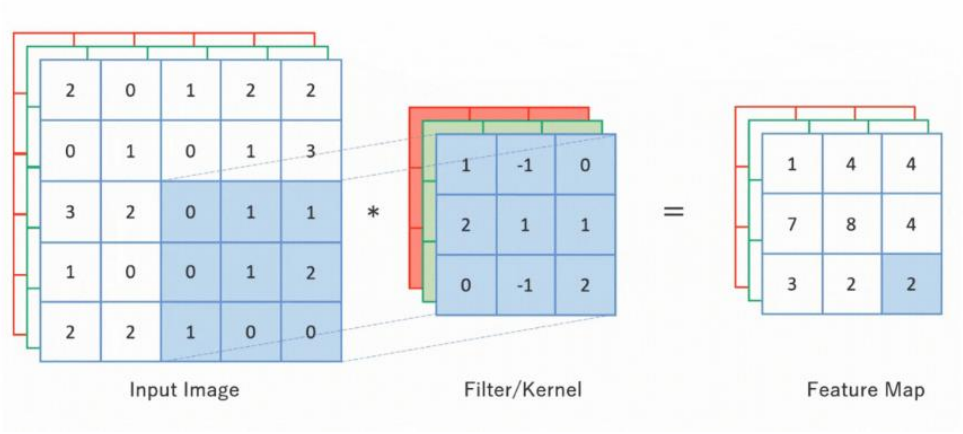
I. Συνελικτικά επίπεδα (convolutional layers)
II. Συγκεντρωτικά επίπεδα (pooling layers)
III. Πλήρως συνδεδεμένα επίπεδα (fully connected layers)



Εικόνα 3.4 – Απεικόνιση της αρχιτεκτονικής ενός CNN [26]

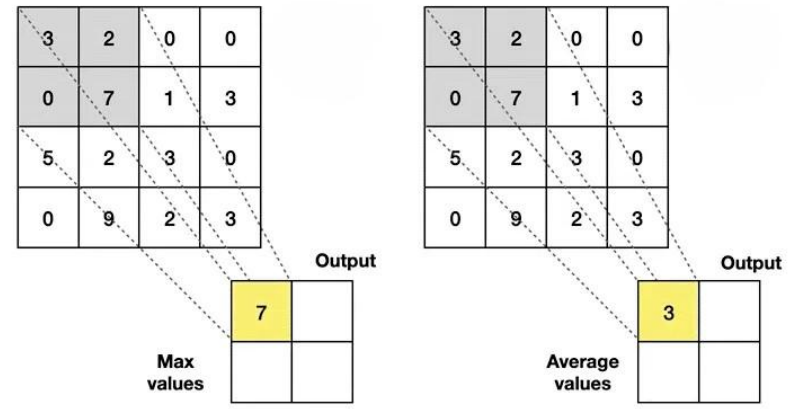
Τα συνελικτικά επίπεδα είναι από τα βασικότερα στρώματα του δικτύου, καθώς σε αυτά τα στρώματα πραγματοποιείται η εξαγωγή χαρακτηριστικών (π.χ. ακμές, γωνίες, υφές) από την αρχική εικόνα, με την εφαρμογή συνελίξεων [27]. Πιο συγκεκριμένα, πραγματοποιείται η εφαρμογή φίλτρων πάνω στην εικόνα εισόδου και παράγεται ένας χάρτης χαρακτηριστικών στην έξοδο, όπως παρουσιάζεται στην Εικόνα 3.5. Η εφαρμογή των φίλτρων πάνω στην εικόνα εισόδου γίνεται με την πράξη της συνέλιξης. Ουσιαστικά το φίλτρο σαρώνει την εικόνα εισόδου και πραγματοποιεί τον υπολογισμό του εσωτερικού γινομένου μεταξύ της εικόνας και του φίλτρου για κάθε εικονοστοιχείο. Το αποτέλεσμα από την συνέλιξη είναι μια πιο αφηρημένη και συμπαγής εκδοχή της αρχικής εικόνας. Αξίζει να σημειωθεί ότι οι παράμετροι των συνελικτικών επιπέδων πρέπει να καθοριστούν πριν από την εκπαίδευση του CNN. Οι παράμετροι αυτές είναι ο αριθμός

των φίλτρων (number of filters), ο οποίος καθορίζει πόσοι θα είναι οι προκύπτοντες χάρτες χαρακτηριστικών, ο διασκελισμός (stride) δηλαδή το βήμα του φίλτρου πάνω στην εικόνα και το zero padding που εφαρμόζεται όταν το φίλτρο και η εικόνα δεν έχουν το απόλυτο ταίριασμα.



Εικόνα 3.5 – Η συνέλιξη μεταξύ εικόνας εισόδου - φίλτρου και ο εξαγώμενος χάρτης χαρακτηριστικών

Τα συγκεντρωτικά επίπεδα βρίσκονται ακριβώς μετά τα συνελικτικά επίπεδα και χρησιμοποιούνται για την μείωση της χωρικής ανάλυσης των χαρτών χαρακτηριστικών. Υπάρχουν δύο τύποι συγκέντρωσης (pooling), το max pooling και το average pooling, όπως παρουσιάζονται στην Εικόνα 3.6. Με χρήση του max pooling, το φίλτρο βρίσκει την μεγαλύτερη τιμή ενός εικονοστοιχείου από μια μικρή περιοχή του χάρτη χαρακτηριστικών και την θέτει ως έξοδο αυτής της περιοχής. Από την άλλη πλευρά, με χρήση του average pooling, το φίλτρο βρίσκει την μέση τιμή των εικονοστοιχείων μιας μικρής περιοχής του χάρτη χαρακτηριστικών και την θέτει ως έξοδο αυτής της περιοχής.



Εικόνα 3.6 – Max pooling vs Average pooling

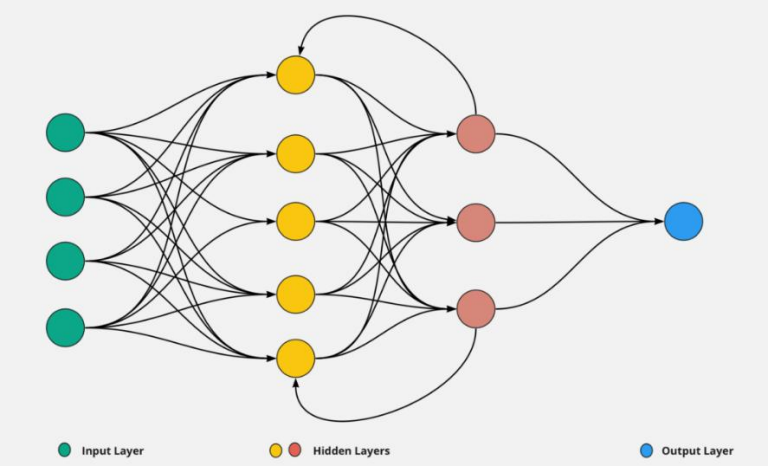
Μετά από τα συνελικτικά και τα συγκεντρωτικά επίπεδα ακολουθούν τα πλήρως συνδεδεμένα στρώματα, τα οποία είναι υπεύθυνα για τις τελικές προβλέψεις. Οι χάρτες χαρακτηριστικών εισέρχονται σε αυτά τα επίπεδα και στην συνέχεια πραγματοποιείται η αντιστοίχισή τους στις τελικές προβλέψεις, μέσω μιας μη γραμμικής συνάρτησης. Η επιλογή της συνάρτησης ενεργοποίησης πρέπει να γίνει με ιδιαίτερη προσοχή, καθώς παίζει σημαντικό ρόλο στην επίδοση του δικτύου. Οι δημοφιλέστερες συναρτήσεις ενεργοποίησης είναι η σιγμοειδής συνάρτηση (Sigmoid), η Tanh, η ReLU και η Softmax. Η επιλογή της συνάρτησης ενεργοποίησης για το επίπεδο εξόδου εξαρτάται από τον τύπο της πρόβλεψης. Αν πρόκειται για ταξινόμηση δυαδικών κατηγοριών (Binary Classification) τότε χρησιμοποιείται η συνάρτηση Softmax ενώ αν πρόκειται για ταξινόμηση πολλαπλών κατηγοριών (Multi-Class Classification) τότε χρησιμοποιείται η συνάρτηση Sigmoid.

Μια παραλλαγή των συνελικτικών νευρωνικών δικτύων είναι τα πλήρως συνελικτικά νευρωνικά δίκτυα (FCNs) [28]. Η κύρια διαφορά τους είναι ότι τα FCNs χρησιμοποιούν συνελικτικά επίπεδα στην θέση των πλήρως συνδεδεμένων στρωμάτων, το οποίο οδηγεί στην επεξεργασία δεδομένων διαφορετικών μεγεθών και την διατήρηση σταθερού αριθμού παραμέτρων, το οποίο δεν συνέβαινε στα CNNs.

3.2.3 Επαναλαμβανόμενα Νευρωνικά Δίκτυα (Recurrent Neural Networks)

Τα επαναλαμβανόμενα νευρωνικά δίκτυα (RNNs) είναι μια κατηγορία των τεχνητών νευρωνικών δικτύων . Στην Εικόνα 3.7 παρουσιάζεται η αρχιτεκτονική ενός

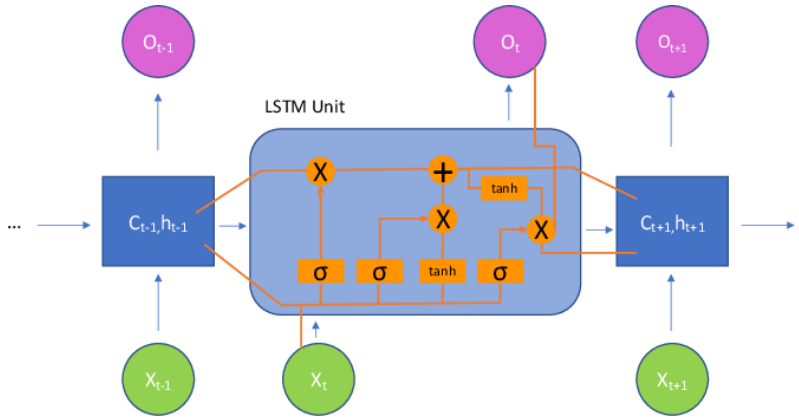
RNN. Το ιδιαίτερο χαρακτηριστικό τους είναι ότι διαθέτουν μια επαναλαμβανόμενη μονάδα (recurrent unit), η οποία προσφέρει την ευελιξία στο δίκτυο να διατηρεί προηγούμενες εισόδους σε μια ακολουθία. Τα RNNs διαφέρουν από τα νευρωνικά δίκτυα εμπρόσθιας τροφοδότησης (feed-forward neural networks), τα οποία είναι ικανά να επεξεργάζονται δεδομένα με ένα πέρασμα από το ένα άκρο του δικτύου στο άλλο, χωρίς να μπορούν να θυμηθούν την προηγούμενη είσοδο που είχαν επεξεργαστεί. Τα RNNs ξεπέρασαν αυτήν την δυσκολία, καθώς με την χρήση επαναλαμβανόμενων μονάδων διατηρούν πληροφορίες από τις προηγούμενες εισόδους και τις ανατροφοδοτούν, με σκοπό την καταγραφή εξαρτήσεων σε βάθος χρόνου. Τα RNNs έχουν ευρεία χρήση σε εργασίες επεξεργασίας χρονοσειρών [29], αναγνώρισης ομιλίας [30][31], η αναγνώριση χειρόγραφου κειμένου [32] κ.α.



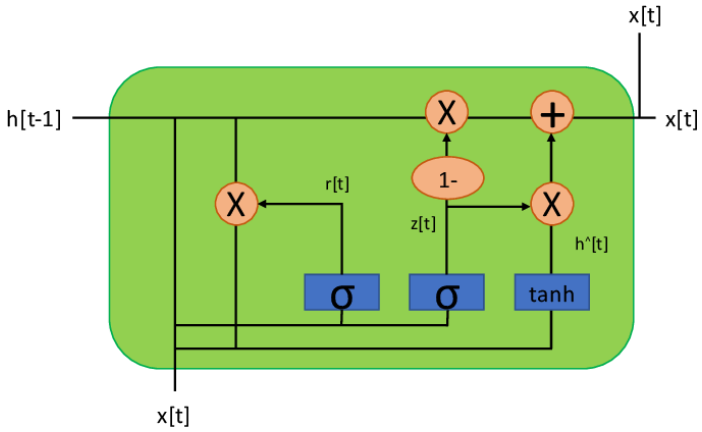
Εικόνα 3.7 – Απεικόνιση της αρχιτεκτονικής ενός RNN [33]

Πρέπει να αναφερθεί ότι έχουν αναπτυχθεί αρκετές παραλλαγές της αρχιτεκτονικής των RNNs, με σκοπό την καταπολέμηση της περιορισμένης ικανότητας των παραδοσιακών RNNs να καταγράφουν μακροπρόθεσμες εξαρτήσεις σε ακολουθίες. Κάποιες από αυτές τις παραλλαγές είναι οι αρχιτεκτονικές Long Short-Term Memory (LSTM) και Gated Recurrent Unit (GRU) [34], οι οποίες παρουσιάζονται στις Εικόνες 3.8 και 3.9 αντίστοιχα. Η αρχιτεκτονική LSTM διαθέτει έναν μηχανισμό μνήμης, με σκοπό την αποθήκευση και την ανάκτηση πληροφοριών στην κρυφή μνήμη, και τρεις πύλες, μια πύλη εισόδου, μια πύλη λήθης και μια πύλη εξόδου. Οι πύλες αυτές είναι υπεύθυνες για τον έλεγχο της ροής των πληροφοριών, με σκοπό την πρόβλεψη της εξόδου. Από την άλλη πλευρά, η αρχιτεκτονική GRU χρησιμοποιεί κρυφές καταστάσεις και διαθέτει

δύο πύλες, μια πύλη επαναφοράς και μια πύλη ενημέρωσης, οι οποίες καθορίζουν πόσες και ποιες πληροφορίες θα διατηρηθούν. Η πύλη ενημέρωσης προκύπτει από τον συνδυασμό της πύλης εισόδου και της πύλης λήθης. Αποτελεί μια απλουστευμένη εκδοχή της αρχιτεκτονικής LSTM και απαιτεί λιγότερες παραμέτρους.



Εικόνα 3.8 – Διάγραμμα μιας μονάδας Long Short-Term Memory (LSTM) [34]

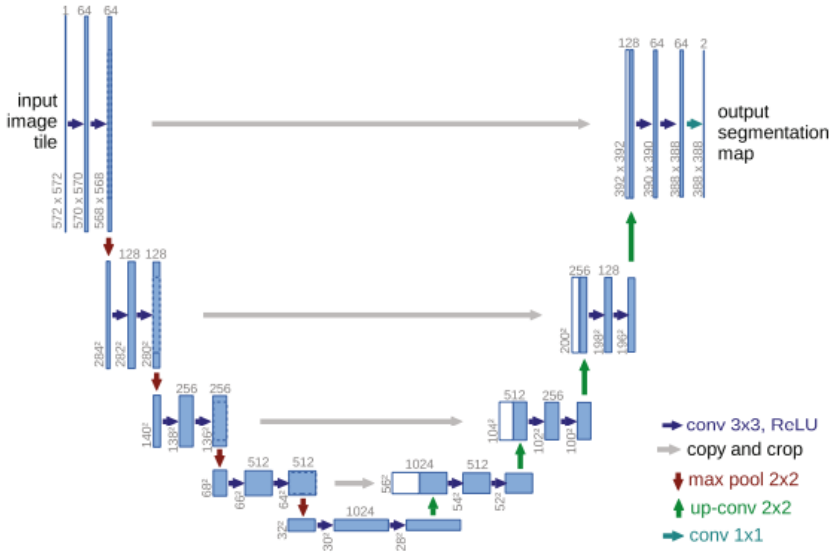


Εικόνα 3.9 – Διάγραμμα μιας μονάδας Gated Recurrent Unit (GRU) [34]

3.3 Παρουσίαση Μοντέλων

3.3.1 Μοντέλο UNet

Το 2015, ο Olaf Ronneberger, ο Philipp Fischer και ο Thomas Brox παρουσίασαν για πρώτη φορά στο Πανεπιστήμιο του Φράιμπουργκ το μοντέλο UNet για την τμηματοποίηση βιοϊατρικών εικόνων [9]. Σήμερα είναι μια από τις πιο ευρέως χρησιμοποιούμενες αρχιτεκτονικές σε οποιαδήποτε εργασία σημασιολογικής τμηματοποίησης, λόγω των εξαιρετικών αποδόσεων που επιτυγχάνει. Η εν λόγω αρχιτεκτονική πήρε το όνομά της από το χαρακτηριστικό της σχήμα, το οποίο μοίαζει με 'U', όπως παρουσιάζεται στην Εικόνα 3.10.



Εικόνα 3.10 – Απεικόνιση της αρχιτεκτονικής του δικτύου UNet

Η αρχιτεκτονική του δικτύου UNet αποτελείται από ένα contracting path, ή αλλιώς encoder, το οποίο φαίνεται στην αριστερή πλευρά του δικτύου, και ένα expanding path, ή αλλιώς decoder, το οποίο φαίνεται στην δεξιά πλευρά του δικτύου. Το contracting path αποτελείται από convolutional layers και max pooling layers, τα οποία είναι υπεύθυνα για την μείωση των διαστάσεων της εικόνας εισόδου και την εξαγωγή των χαρακτηριστικών της. Κάθε convolutional layer ακολουθείται από μια συνάρτηση ενεργοποίησης ReLU. Η συνάρτηση ReLU εισάγει την μη γραμμικότητα στο δίκτυο, το οποίο βοηθάει στην καλύτερη γενίκευση των δεδομένων εκπαίδευσης. Αντίστοιχα, το expanding path αποτελείται από upsampling layers και convolutional layers, τα οποία

βοηθούν στην αύξηση των διαστάσεων της εικόνας και στην επίτευξη της κατάτμησης, μέσω των χαρακτηριστικών που έχουν εξαχθεί από τα προηγούμενα επίπεδα. Το U-Net διαθέτει ένα ειδικό τμήμα, το οποίο ονομάζεται skip connection. Σε αυτό το τμήμα, πραγματοποιείται η διατήρηση των πληροφοριών που χάνονται στο contracting path, το οποίο βοηθά τα επίπεδα του expanding path να εντοπίσουν τα χαρακτηριστικά με μεγαλύτερη ακρίβεια, καθιστώντας το U-Net ένα πολύ ισχυρό μοντέλο για εργασίες τμηματοποίησης εικόνων.

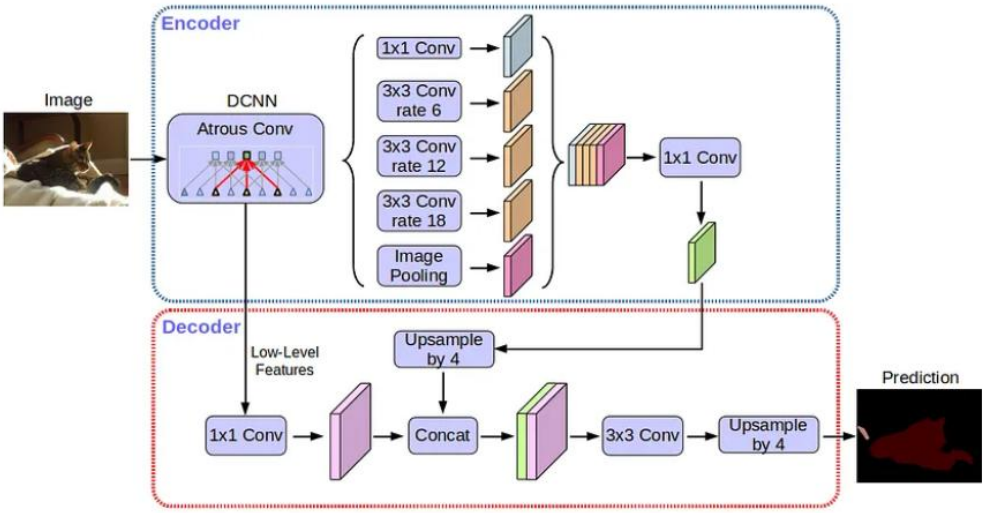
Επιπλέον, πρέπει να αναφερθεί ότι, κατά τη διάρκεια του contracting path, η εικόνα εισόδου μειώνεται διαδοχικά σε ύψος και πλάτος, αλλά αυξάνεται ο αριθμός των καναλιών. Η αύξηση στα κανάλια δίνει την δυνατότητα στο δίκτυο να αναγνωρίζει χαρακτηριστικά υψηλού επιπέδου, καθώς προχωρά στη διαδρομή. Στο τέλος του contracting path, υπάρχει ένα επίπεδο γνωστό ως bottleneck layer, το οποίο αποτελεί το βαθύτερο πεδίο του δικτύου. Στο bottleneck layer, δημιουργείται ένας χάρτης χαρακτηριστικών. Στη συνέχεια, το expanding path λαμβάνει τον χάρτη χαρακτηριστικών από το bottleneck layer και τον μετατρέπει σε μια εικόνα του ίδιου μεγέθους με την αρχική. Αυτό πραγματοποιείται με την χρήση upsampling layers, τα οποία μειώνουν τον αριθμό των καναλιών. Τέλος, κάθε pixel στην εικόνα εξόδου αντιπροσωπεύει μια ετικέτα που αντιστοιχεί σε μια συγκεκριμένη κατηγορία ή αντικείμενο της εικόνας εισόδου.

Πιο αναλυτικά, το πρώτο βήμα είναι η είσοδος της εικόνας στο δίκτυο. Αμέσως μετά, στο contracting path, εφαρμόζονται δύο διαδοχικές συνελίξεις 3x3 και ένα max pooling επίπεδο 2x2 με βηματισμό 2. Με την ολοκλήρωση αυτής της διαδικασίας, διπλασιάζονται τα κανάλια χαρακτηριστικών και υποδιπλασιάζεται το μέγεθος της εικόνας. Στο expanding path, εφαρμόζονται μια συνέλιξη 2x2, με σκοπό την μείωση του αριθμού των καναλιών στα μισά, και δύο διαδοχικές συνελίξεις 3x3. Στο τελευταίο επίπεδο, εφαρμόζεται μια συνέλιξη 1x1 για την αντιστοίχιση ενός διανύσματος χαρακτηριστικών σε κάθε εικονοστοιχείο, με σκοπό την δημιουργία μιας μάσκας κατάτμησης για κάθε κατηγορία.

Πρέπει να αναφερθεί ότι, στην περίπτωση που το μοντέλο οδηγηθεί σε υπερπροσαρμογή (overfitting), τότε μπορούμε να χρησιμοποιήσουμε dropout επίπεδα στη διαδρομή bridge – bottleneck, με σκοπό την αντιμετώπιση αυτού του φαινομένου. Τα dropout επίπεδα απενεργοποιούν τυχαία νευρώνες κατά τη διάρκεια της εκπαίδευσης, και με αυτόν τον τρόπο το μοντέλο σταματάει να βασίζεται στις εξόδους συγκεκριμένων νευρώνων, αλλά βασίζεται σε μεγαλύτερο φάσμα αυτών. Η τιμή του μπορεί να κυμαίνεται από 0 έως 1. Για παράδειγμα, αν η τιμή του είναι 0.1, τότε απενεργοποιείται το 10% των νευρώνων σε κάθε επίπεδο.

3.3.2 Μοντέλο DeepLab

Το μοντέλο DeepLab αποτελεί επίσης ένα αξιοσημείωτο μοντέλο που χρησιμοποιείται σε εργασίες σημασιολογικής τμηματοποίησης εικόνων. Το εν λόγω μοντέλο σχεδιάστηκε και παρουσιάστηκε από την Google. Με την πάροδο των χρόνων, παρουσιάστηκαν ορισμένες βελτιώσεις του μοντέλου, το οποίο οδήγησε στις νεότερες εκδόσεις του DeepLab [16][17][18][19]. Το DeepLabV1 αποτελεί μια εξέλιξη των FCN, κάθως οι βελτιώσεις που πραγματοποιήθηκαν σε αυτά τα μοντέλα οδήγησαν στην επιτυχημένη πορεία του DeepLabV1 σε εργασίες σημασιολογικής τμηματοποίησης [16]. Τόσο το DeepLabV1 όσο και το DeepLabV2 χρησιμοποιούν βαθιά συνελικτικά νευρωνικά δίκτυα με atrous convolutions και πεδίο CRF. Η διαφορά μεταξύ τους είναι ότι το DeepLabV2 χρησιμοποιεί Atrous Spatial Pyramid Pooling και εφαρμόζει πιο σύγχρονα βαθια συνελικτικά νευρωνικά δίκτυα. Ένα ακόμη πιο βελτιωμένο μοντέλο είναι το DeepLabV3, το οποίο δεν χρησιμοποιεί πεδίο CRF και παρουσιάζει καλύτερα αποτελέσματα από τις προηγούμενες εκδόσεις του DeepLab. Η επέκταση του DeepLabV3 και η πιο σύγχρονη έκδοση του DeepLab είναι το DeepLabV3+, το οποίο αποτελείται από μια αρχιτεκτονική κωδικοποιητή-αποκωδικοποιητή και από Atrous Spatial Pyramid Pooling [17][19]. Η χρήση κωδικοποιητή-αποκωδικοποιητή οδηγεί σε αύξηση της απόδοσης της τμηματοποίησης όσον αφορά τα περιγράμματα των αντικειμένων. Η αρχιτεκτονική του μοντέλου DeepLabV3+ παρουσιάζεται στην Εικόνα 3.11.



Εικόνα 3.11 – Απεικόνιση της αρχιτεκτονικής του δικτύου DeepLabV3+

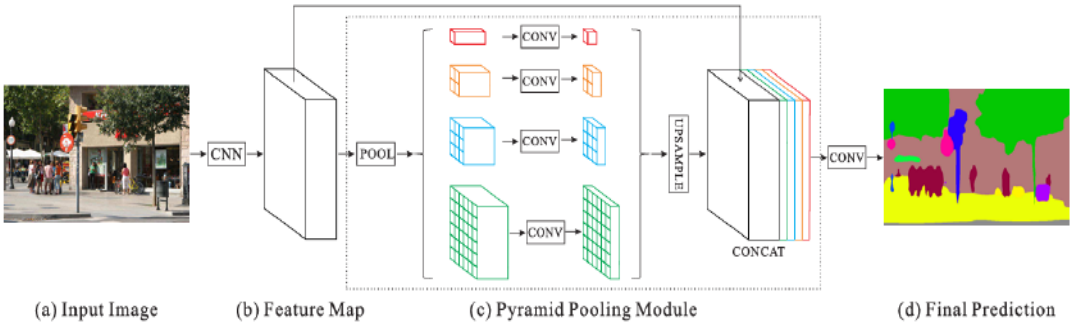
Το DeepLabV3+ δεν λειτουργεί με τον ίδιο τρόπο που λειτουργούν άλλες αρχιτεκτονικές. Η αρχιτεκτονική του μοντέλου DeepLabV3+ μπορεί να διαθέτει ένα δίκτυο κορμού (backbone network), το οποίο μπορεί να είναι ένα ResNet ή ένα MobileNet. Το συγκεκριμένο δίκτυο είναι υπεύθυνο για την εξαγωγή των χαρτών χαρακτηριστικών, δηλαδή την αρχική αναπαράσταση των χαρακτηριστικών της εικόνας που εισάγεται στο δίκτυο. Η επιλογή του δικτύου πρέπει να γίνει με ιδιαίτερη προσοχή, καθώς μπορεί να επηρεάσει σε μεγάλο βαθμό την απόδοση του μοντέλου. Στην συνέχεια, οι χάρτες χαρακτηριστικών επεξεργάζονται μέσω atrous convolutions με διαφορετικούς ρυθμούς, με σκοπό την καταγραφή πληροφοριών πολλαπλής κλίμακας. Οι atrous convolutions, γνωστές και ως dilated convolutions, βοηθούν στον έλεγχο της ανάλυσης υπολογισμού των χαρακτηριστικών και την μεγέθυνση του οπτικού πεδίου των φίλτρων, χωρίς την αύξηση των παραμέτρων ή του υπολογιστικού όγκου. Αμέσως μετά, γίνεται χρήση του Atrous Spatial Pyramid Pooling (ASPP), μέσω του οποίου εκτελούνται παράλληλα πολλαπλές atrous convolutions με διαφορετικούς ρυθμούς, με σκοπό την να ληφθούν υπόψη διαφορετικές κλίμακες για αποδοτικότερη τμηματοποίηση. Τελαυταίο βήμα είναι η αναβάθμιση των χαρτών των τελικών χαρακτηριστικών, με σκοπό να ταιριάζουν με την ανάλυση της αρχικής εικόνας και η χρήση τους για την δημιουργία του τελικού χάρτη τμηματοποίησης βάσει εικονοστοιχείου. Προαιρετικά μπορεί να χρησιμοποιηθεί ένα Conditional Random Field (CRF), για την βελτίωση των αποτελεσμάτων της τμηματοποίησης.

3.3.3 Μοντέλο PSPNet

Το μοντέλο PSPNet είναι ακόμη ένα μοντέλο σημασιολογικής τμηματοποίησης εικόνων σε επίπεδο εικονοστοιχείου, το οποίο σχεδιάστηκε και αναπτύχθηκε το 2017, από τον Hengshuang Zhao και άλλους ερευνητές του πανεπιστημίου του Hong Kong [15]. Ιδιαίτερο χαρακτηριστικό της αρχιτεκτονικής του PSPNet είναι η ενσωμάτωση μιας μονάδας pyramid pooling (PPM), η οποία προσφέρει στο δίκτυο την δυνατότα να καταγράφει τοπικές και παγκόσμιες πληροφορίες σε διαφορετικές κλίμακες, με σκοπό μια ακριβέστερη τμηματοποίηση.

Η αρχιτεκτονική του μοντέλου PSPNet παρουσιάζεται στην Εικόνα 3.12. Το πρώτο βήμα είναι η εισαγωγή της εικόνας στο δίκτυο και η εξαγωγή χαρακτηριστικών με χρήση ενός προεκπαιδευμένου μοντέλου. Στην συνέχεια, πραγματοποιείται η συλλογή πληροφοριών από τον χάρτη χαρακτηριστικών, μέσω μιας μονάδας Pyramid Pooling. Πιο συγκεκριμένα, η μονάδα pyramid pooling διαθέτει μια πυραμίδα τεσσάρων κλιμάκων

(1x1, 2x2, 3x3, 6x6), η οποία καλύπτει όλη την εικόνα. Αυτά τα μεγαλύτερα παράθυρα χρησιμοποιούνται για την κατανόηση όλης της σκηνής. Αμέσως μετά, ακολουθεί η συγχώνευση των χαρακτηριστικών που εξάγονται από τις διαφορετικές κλίμακες και η σύνδεσή τους με τον αρχικό χάρτη χαρακτηριστικών, ο οποίος προέκυψε από το προεκπαιδευμένο δίκτυο. Τέλος, οι πληροφορίες επεξεργάζονται μέσω ενός συνελικτικού επιπέδου, με σκοπό την δημιουργία του τελικού χάρτη τμηματοποίησης.



Εικόνα 3.12 – Απεικόνιση της αρχιτεκτονικής του δικτύου PSPNet

3.4 Παράμετροι Εκπαίδευσης (Hyperparameters)

Η διαδικασία ρύθμισης των παραμέτρων εκπαίδευσης σε ένα νευρωνικό δίκτυο είναι εξαιρετικά σημαντική. Με την σωστή επιλογή τους, το δίκτυο μπορεί να οδηγηθεί στην επίτευξη υψηλών αποδόσεων. Στην συνέχεια, παρουσιάζονται ο αριθμός των εποχών (epochs), το μέγεθος παρτίδας (batch size), ο ρυθμός εκμάθησης (learning rate), τα είδη των βελτιστοποιητών (optimizers) και τα είδη των συναρτήσεων κόστους (loss functions).

3.4.1 Αριθμός Εποχών (Epochs)

Μια από τις παραμέτρους εκπαίδευσης του δικτύου είναι ο αριθμός των εποχών (epochs), ο οποίος αντιπροσωπεύει τον αριθμό των φορών που το μοντέλο θα εκτεθεί σε όλα τα δεδομένα εκπαίδευσης. Για μεγάλο αριθμό εποχών, δηλαδή αν το μοντέλο εκτεθεί αρκετές φορές στα δεδομένα εκπαίδευσης, υπάρχει μεγάλη πιθανότητα να οδηγηθεί σε υπερπροσαρμογή, με αποτέλεσμα να εμφανίζει χαμηλή απόδοση.

3.4.2 Μέγεθος Παρτίδας (Batch Size)

Το μέγεθος παρτίδας (batch size) αποτελεί μια παράμετρο εκπαίδευσης, η οποία καθορίζει τον αριθμό των επαναλήψεων ανά εποχή. Το batch size επηρεάζει την ταχύτητα και την αποτελεσματικότητα της εκπαίδευσης. Μια μικρή τιμή του batch size έχει ως αποτέλεσμα μια πιο αργή εκπαίδευση αλλά καλύτερη γενίκευση από το μοντέλο ενώ μια μεγάλη τιμή του batch size έχει ως αποτέλεσμα μια πιο γρήγορη εκπαίδευση με την απαίτηση περισσότερης υπολογιστικής μνήμης.

3.4.3 Ρυθμός Εκμάθησης (Learning Rate)

Ο ρυθμός εκμάθησης (learning rate) αποτελεί μια παράμετρο εκπαίδευσης του δικτύου. Η εύρεση της βέλτιστης τιμής για το learning rate είναι σημαντική, καθώς καθορίζει το μέγεθος του βήματος, με σκοπό το μοντέλο να συγκλίνει στην ελάχιστη τιμή μιας συνάρτησης κόστους. Η τιμή του learning rate είναι σχετικά μικρή και κυμαίνεται μεταξύ 0 και 1.

3.4.4 Βελτιστοποιητές (Optimizers)

Οι βελτιστοποιητές αποτελούν αλγορίθμους, οι οποίοι είναι υπεύθυνοι για την τροποποίηση των βαρών του δικτύου, με σκοπό την ελαχιστοποίηση της συνάρτησης κόστους. Οι πιο ευρέως χρησιμοποιούμενοι βελτιστοποιητές είναι: ο Adam, ο Stochastic Gradient Descent (SGD), ο RMSprop, ο Adagrad κ.α. Πρέπει να σημειωθεί ότι οι βασικές διαφορές μεταξύ αυτών των βελτιστοποιητών ανιχνεύονται στους αλγορίθμους, οι οποίοι χρησιμοποιούνται τόσο για την ενημέρωση του ρυθμού εκμάθησης, όσο και για την ενημέρωση των παραμέτρων του μοντέλου. Για παράδειγμα, ο Adam και ο RMSprop χρησιμοποιούν πιο πολύπλοκους αλγόριθμους, οι οποίοι αν και προσφέρουν περισσότερα οφέλη, μπορεί να χρειάζονται πιο προσεκτική ρύθμιση. Από την άλλη, ο SGD και ο Adagrad χρησιμοποιούν πιο απλούς αλγόριθμους.

3.4.5 Συναρτήσεις Κόστους (Loss Functions)

Οι συναρτήσεις κόστους είναι υπεύθυνες για τον υπολογισμό της απόκλισης των προβλέψεων του δικτύου από τις πραγματικές τιμές. Η σωστή επιλογή της συνάρτησης κόστους είναι σημαντική για την επίδοση του δικτύου. Υπάρχει πληθώρα συναρτήσεων

κόστους, όπως Binary Cross Entropy, Categorical Cross Entropy, Dice Loss κ.α. Η Binary Cross Entropy χρησιμοποιείται όταν υπάρχουν δύο κατηγορίες ενώ σε περίπτωση ύπαρξης πολλαπλών κατηγοριών, χρησιμοποιείται η Categorical Cross Entropy. Η BCE εξάγει μια πρόβλεψη με τιμή πιθανότητας μεταξύ 0 και 1. Το 0 αντιστοιχεί στην πρώτη κατηγορία ενώ το 1 αντιστοιχεί στην δεύτερη. Το χαρακτηριστικό της Dice Loss είναι η ακριβής ανίχνευση των ορίων και σε συνδυασμό με την BCE, η οποία μετράει την ομοιότητα μεταξύ των εικονοστοιχείων της πραγματικής και της προβλεπόμενης μάσκας, προσφέρεται μια ισορροπημένη προσέγγιση στην τμηματοποίηση εικόνων.

3.5 Μέθοδοι Αξιολόγησης

Πριν την διαδικασία της εκπαίδευσης, είναι σημαντικό να πραγματοποιηθεί η επιλογή των σωστών μεθόδων αξιολόγησης, που θα χρησιμοποιηθούν για τον έλεγχο της απόδοσης του μοντέλου. Στην συνέχεια, παρουσιάζονται ορισμένες μετρικές αξιολόγησης, όπως Intersection of Union (IoU), Dice Coefficient, Pixel Accuracy, Precision, Recall και F1-score.

3.5.1 Intersection of Union (IoU)

Η μετρική Intersection of Union, η οποία είναι γνωστή και ως Jaccard Index, χρησιμοποιείται ευρέως για την αξιολόγηση των μοντέλων σε εργασίες ανίχνευσης αντικειμένων και τμηματοποίησης εικόνων. Η εν λόγω μετρική περιγράφεται από την σχέση 3.1. Ουσιαστικά πρόκειται για την τομή διαιρεμένη με την ένωση του συνόλου των δειγμάτων. Αφού ολοκληρωθεί η εκπαίδευση του μοντέλου, ακολουθεί η σύγκριση των προβλεπόμενων μασκών με τις πραγματικές μάσκες (ground truth) και τέλος πραγματοποιείται ο υπολογισμός της περιοχής της επικάλυψης και της περιοχής της ένωσης ανάμεσα στις πραγματικές περιοχές και στις περιοχές των προβλέψεων.

$$\text{IoU} = \frac{|A \cap B|}{|A \cup B|} \qquad (3.1)$$

Η παραπάνω διαδικασία πραγματοποιείται για κάθε εικόνα και το τελικό αποτέλεσμα για το σύνολο δοκιμών προκύπτει από τον μέσο όρο των τιμών που υπολογίστηκαν. Μπορεί

να θεωρηθεί ότι το μοντέλο κάνει καλές προβλέψεις για τιμές του Intersection of Union μεγαλύτερες από 0.5. Στην Εικόνα 3.13 παρουσιάζονται ορισμένα παραδείγματα κακών, καλών και εξαιρετικών τιμών για την μετρική IoU.

Εικόνα 3.13 – Απεικόνιση διάφορων περιπτώσεων για την τιμή του IoU

3.5.2 Dice Coefficient

Ο Dice Coefficient αποτελεί μια μετρική αξιολόγησης, η οποία υπολογίζει την επικάλυψη μεταξύ της προβλεπόμενης και της πραγματικής περιοχής. Περιγράφεται από την σχέση 3.2 και υπολογίζεται ως το διπλάσιο της τομής των προβλεπόμενων και των πραγματικών περιοχών, διαιρεμένο με το αθροισμά τους. Το εύρος τιμών που λαμβάνει ο Dice Coefficient είναι από 0 έως 1. Για τιμή του Dice Coefficient ίση με 1, έχουμε την τέλεια επικάλυψη μεταξύ της προβλεπόμενης και της πραγματικής περιοχής.

$$DC = 2 \times \frac{|A \cap B|}{|A| + |B|} \qquad (3.2)$$

3.5.3 Pixel Accuracy

Η ακρίβεια εικονοστοιχείου είναι ακόμη μια μετρική αξιολόγησης και περιγράφεται από την σχέση 3.3. Ουσιαστικά πρόκειται για τον αριθμό των σωστά ταξινομημένων εικονοστοιχείων, δηλαδή το άθροισμα των πραγματικά θετικών (TP) και των πραγματικά

αρνητικών (TN) εικονοστοιχείων, διαιρεμένο με τον συνολικό αριθμό των εικονοστοιχείων, δηλαδή το άθροισμα των TP, TN, FP και FN. Η εν λόγω μετρική δεν προσφέρει αξιοπιστία καθώς ο μεγαλύτερος αριθμός των εικονοστοιχείων βρίσκεται στο background και όχι στην επιθυμητή περιοχή της εικόνας, η οποία αποτελείται από μικρότερο αριθμό εικονοστοιχείων.

$$\text{Pixel Accuracy} = \frac{\text{Αριθμός σωστά ταξινομημένων εικονοστοιχείων}}{\text{Συνολικός αριθμός εικονοστοιχείων}} \qquad (3.3)$$

3.5.4 Precision

Η μετρική Precision είναι το ποσοστό όλων των ταξινομήσεων του μοντέλου ως θετικές, οι οποίες είναι πραγματικά θετικές. Η εν λόγω μετρική περιγράφεται από την σχέση 3.4. Αν ένα μοντέλο εμφάνιζε precision ίση με 1 τότε θα μιλούσαμε για ένα υποθετικά τέλειο μοντέλο, του οποίου οι ψευδώς θετικές (FP) ταξινομήσεις θα ήταν μηδέν. Αν το σύνολο δεδομένων παρουσιάζει ανισορροπία, δηλαδή ο αριθμός των πραγματικά θετικών (TP) είναι πολύ μικρός τότε η ανάκληση δεν είναι τόσο χρήσιμη σαν μέτρηση. Μείωση των ψευδών θετικών (FP) οδηγεί σε βελτίωση της μετρικής Precision.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \qquad (3.4)$$

3.5.5 Recall

Η ανάκληση (recall), η οποία είναι γνωστή και ως true positive rate (TPR), είναι το ποσοστό όλων των πραγματικών θετικών που ταξινομήθηκαν σωστά ως θετικά. Η ανάκληση περιγράφεται από την σχέση 3.5. Ο παρανομαστής του κλάσματος αποτελείται από τις πραγματικά θετικές (TP) και από τις ψευδώς αρνητικές (FN), οι οποίες είναι στην ουσία πραγματικά θετικές αλλά ταξινομήθηκαν εσφαλμένα ως

αρνητικές. Αν ένα μοντέλο εμφάνιζε ανάκληση ίση με 1 τότε θα μιλούσαμε για ένα υποθετικά τέλειο μοντέλο, το οποίο θα έχει ποσοστό ανίχνευσης 100%. Σε αυτήν την περίπτωση οι ψευδώς αρνητικές θα ήταν μηδέν. Η ανάκληση δεν είναι τόσο χρήσιμη σαν μέτρηση στην περίπτωση που το σύνολο δεδομένων παρουσιάζει ανισορροπία, δηλαδή ο αριθμός των πραγματικά θετικών (TP) είναι πολύ μικρός. Μείωση των ψευδών αρνητικών (FN) οδηγεί σε βελτίωση της ανάκλησης.

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \qquad (3.5)$$

Αξίζει να σημειωθεί ότι οι δύο τελευταίες μετρικές Precision και Recall έχουν αντιστρόφως ανάλογη σχέση, δηλαδή η βελτίωση της μιας οδηγεί σε χειρότερα αποτελέσματα για την άλλη. Επιπλέον αυτές οι δύο μετρικές αξιολόγησης χρησιμοποιούνται για τον υπολογισμό της επόμενης μετρικής F1-score.

3.5.6 F1-score

Όπως προαναφέρθηκε, η μετρική F1-score προκύπτει από τον συνδυασμό των μετρικών Precision και Recall. Η εν λόγω μετρική περιγράφεται από την σχέση 3.6 και λαμβάνει τιμές μεταξύ 0 και 1. Υψηλή τιμή για την μετρική F1-score, υποδηλώνει την ισορροπία μεταξύ των μετρικών Precision και Recall. Χαμηλή τιμή για την μετρική F1-score, υποδηλώνει ότι το δίκτυο είτε παρουσιάζει χαμηλή τιμή για την μετρική Precision είτε παρουσιάζει χαμηλή τιμή για την μετρική Recall.

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \qquad (3.6)$$

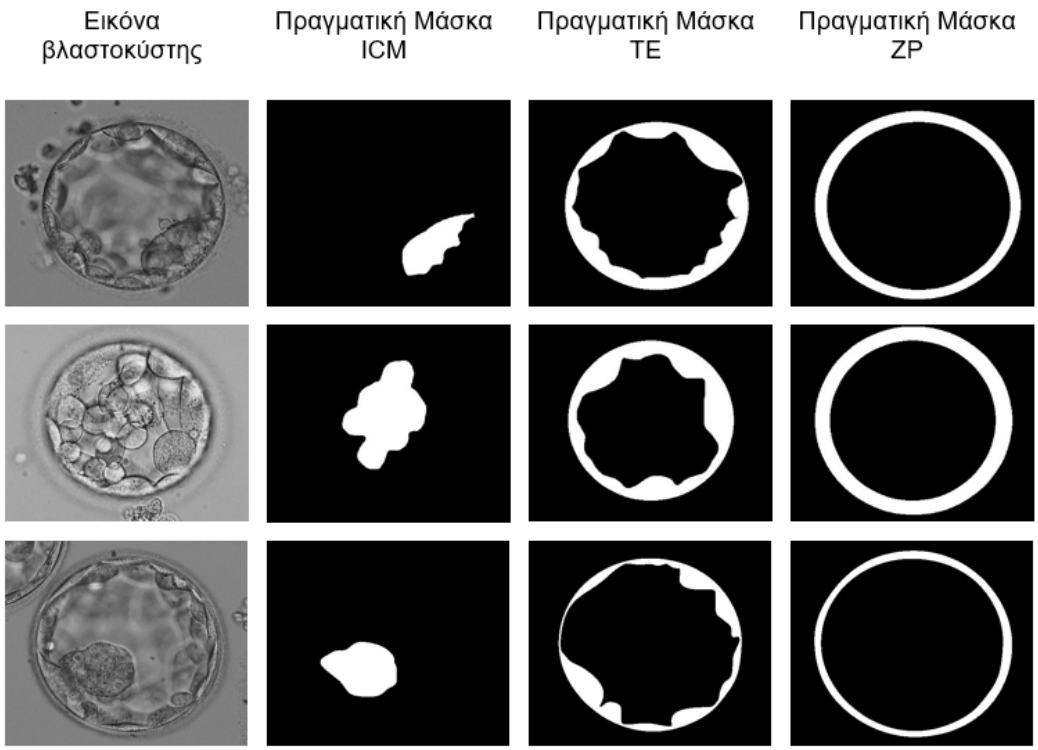
4. Μεθοδολογία Έρευνας

4.1 Εισαγωγή

Στο κεφάλαιο 4 περιγράφεται η μεθοδολογία που ακολουθήθηκε, με σκοπό την επίτευξη του στόχου της εργασίας. Αρχικά, παρουσιάζεται η διαδικασία της συλλογής και της προετοιμασίας των δεδομένων πριν την εισαγωγή τους στα μοντέλα, καθώς και η εισαγωγή των απαραίτητων βιβλιοθηκών που χρησιμοποιήθηκαν. Στην συνέχεια, περιγράφεται η υλοποίηση κάθε μοντέλου ξεχωριστά και αναφέρονται οι υπερπαράμετροι εκπαίδευσης που χρησιμοποιήθηκαν για τα εν λόγω μοντέλα. Πρέπει να σημειωθεί ότι, για την επιλογή των τιμών των υπερπαραμέτρων εκπαίδευσης, πραγματοποιήθηκε πειραματικός έλεγχος για την εύρεση του κατάλληλου αριθμού των εποχών (epochs), του κατάλληλου μεγέθους των παρτίδων (batch size) και του κατάλληλου ρυθμού εκμάθησης (learning rate), με σκοπό τα δίκτυα να εμφανίζουν την μέγιστη δυνατή απόδοση. Τέλος, περιγράφεται η διαδικασία της εκπαίδευσης και των προβλέψεων των μοντέλων, καθώς και ποιές μετρικές αξιολόγησης χρησιμοποιήθηκαν.

4.2 Συλλογή Δεδομένων

Για την εκτέλεση της συγκεκριμένης εργασίας, ήταν απαραίτητη η συλλογή του σύνολου δεδομένων από http://vault.sfu.ca/index.php/s/066vGJfviJMYuP6. Πρέπει να σημειωθεί ότι το εν λόγω σύνολο δεδομένων έχει εγκριθεί από την Καναδική Επιτροπή Δεοντολογίας της Έρευνας στις 24 Μαΐου 2017 και είναι διαθέσιμο στην επιστημονική κοινότητα κατόπιν αιτήματος. Το σύνολο δεδομένων αποτελείται από 249 εικόνες βλαστοκύστεων και 249 μάσκες για κάθε ένα από τα τρία χαρακτηριστικά ICM, TE, ZP των βλαστοκύστεων. Στην Εικόνα 4.1 παρουσιάζεται ένα δείγμα των εικόνων που συλλέχθηκαν, πριν την εισαγωγή τους στο δίκτυο.



Εικόνα 4.1 – Δείγμα εικόνων βλαστοκύστεων και πραγματικών μασκών για ICM, TE & ZP

4.3 Εισαγωγή Βιβλιοθηκών

Για την λειτουργία του κώδικα, ήταν απαραίτητη η εισαγωγή ορισμένων βιβλιοθηκών, όπως παρουσιάζεται στην Εικόνα 4.2. Πιο συγκεκριμένα, έγινε χρήση της βιβλιοθήκης os, τόσο για την ανάγνωση αρχείων από τους φακέλους, όσο και για την δημιουργία φακέλων, με σκόπο την αποθήκευση των προβλεπόμενων μασκών. Επιπλέον, έγινε χρήση της βιβλιοθήκης cv2 για την φόρτωση των εικόνων των βλαστοκύστεων, την φόρτωση των πραγματικών μασκών σε κλίμακα grayscale, την αλλαγή του μεγέθους των εικόνων, και την αποθήκευση των προβλεπόμενων μασκών. Η βιβλιοθήκη Numpy χρησιμοποιήθηκε για την δημιουργία πινάκων, με σκοπό την αποθήκευση τόσο των εικόνων των βλαστοκύστεων όσο και των αντίστοιχων μασκών τους. Από την άλλη, η βιβλιοθήκη Matplotlib χρησιμοποιήθηκε τόσο για την απεικόνιση των εικόνων των βλαστοκύστεων και των προβλεπόμενων μασκών, όσο και για την δημιουργία των διαγραμμάτων των μετρικών αξιολόγησης. Για τους υπολογισμούς τόσο των μετρικών αξιολόγησης όσο και των συναρτήσεων κόστους, ήταν απαραίτητη η χρήση του module tensorflow.keras.backend της βιβλιοθήκης Tensorflow, με την

συντομογραφία K. Επίσης, χρησιμοποιήθηκαν τα modules tensorflow.keras.losses, tensorflow.keras.Input, tensorflow.keras.layers και tensorflow.keras.Model από την βιβλιοθήκη Tensorflow, για την υλοποίηση και την εκπαίδευση των μοντέλων. Τέλος, χρησιμοποιείται το module sklearn.model_selection της βιβλιοθήκης scikit-learn για τον διαχωρισμό των δεδομένων σε σύνολα εκπαίδευσης και δοκιμής.

```
import os
import cv2
import numpy as np
import matplotlib.pyplot as plt
import tensorflow.keras.backend as K
from tensorflow.keras.optimizers import Adam
from tensorflow.keras import losses, Input, layers, Model
from sklearn.model_selection import train_test_split
```

Εικόνα 4.2 – Εισαγωγή Βιβλιοθηκών

4.4 Προετοιμασία Δεδομένων

Ένα από τα σημαντικότερα σημεία σε τέτοιου είδους προβλήματα είναι η σωστή προετοιμασία των δεδομένων πριν την εισαγωγή τους στο μοντέλο, ώστε να είναι αποτελεσματική η εκπαίδευση και η αξιολόγησή του. Αρχικά, δημιουργήθηκαν δύο συναρτήσεις για την φόρτωση των εικόνων και των μασκών. Η πρώτη συνάρτηση, με το όνομα load_images() φορτώνει όλες τις εικόνες που βρίσκονται σε έναν συγκεκριμένο φάκελο, ο οποίος έχει προκαθοριστεί, αλλάζει το μέγεθος των εικόνων σε 128x128, και κατόπιν τις αποθηκεύει σε έναν πίνακα με το όνομα images. Η δεύτερη συνάρτηση, load_masks() φορτώνει τις μάσκες που βρίσκονται σε έναν άλλο προκαθορισμένο φάκελο, αλλάζει τις διαστάσεις των μασκών σε 128x128, και τις αποθηκεύει σε έναν πίνακα με όνομα masks. Αξίζει να σημειωθεί ότι η αλλαγή του μεγέθους των εικόνων σε 128x128 έγινε με σκοπό όλες οι εικόνες να έχουν το ίδιο μέγεθος κατά την εισαγωγή τους στο δίκτυο. Επιπλέον, πρέπει να σημειωθεί ότι η φόρτωση των εικόνων των βλαστοκύστεων γίνεται ως εικόνες με τρία κανάλια χρώματος (RGB), ενώ η φόρτωση των εικόνων των μασκών γίνεται ως εικόνες με ένα κανάλι, δηλαδή εικόνες σε κλίμακα του γκρι (grayscale). Επομένως, η διάσταση των εικόνων των βλαστοκύστεων θα είναι 128x128x3, ενώ η διάσταση των εικόνων των μασκών θα είναι 128x128x1. Τέλος, επειδή οι εικόνες αποθηκεύονται με τιμές των pixels στο διάστημα [0, 255], κρίνεται απαραίτητη η κανονικοποίηση αυτών των τιμών στο διάστημα [0, 1], καθώς βελτιστοποιείται η

απόδοση του μοντέλου. Στην Εικόνα 4.3 παρουσιάζεται ο κώδικας για την προετοιμασία των δεδομένων.

```
# Διαδρομές για τις εικόνες και τις μάσκες
image_path = r'C:\Full\Path\To\Your\Images\Folder'
icm_mask_path = r'C:\Full\Path\To\Your\GT_ICM\Folder'
te_mask_path = r'C:\Full\Path\To\Your\GT_TE\Folder'
zp_mask_path = r'C:\Full\Path\To\Your\GT_ZP\Folder'

# Φόρτωση εικόνων
def load_images(image_dir):
    images = []
    for filename in os.listdir(image_dir):
        img = cv2.imread(os.path.join(image_dir, filename))
        if img is not None:
            img = cv2.resize(img, (128, 128))  # Αλλαγή μεγέθους των εικόνων
            images.append(img)
    return np.array(images)

# Φόρτωση μασκών ως μονοκάναλες εικόνες
def load_masks(mask_dir):
    masks = []
    for filename in os.listdir(mask_dir):
        mask = cv2.imread(os.path.join(mask_dir, filename), cv2.IMREAD_GRAYSCALE)
        if mask is not None:
            mask = cv2.resize(mask, (128, 128))  # Αλλαγή μεγέθους των μασκών
            mask = mask[..., np.newaxis]
            masks.append(mask)
    return np.array(masks)

# Φόρτωση εικόνων και μασκών
images = load_images(image_path)
masks_icm = load_masks(icm_mask_path)
masks_te = load_masks(te_mask_path)
masks_zp = load_masks(zp_mask_path)

# Κανονικοποίηση
images = images / 255.0
masks_icm = masks_icm / 255.0
masks_te = masks_te / 255.0
masks_zp = masks_zp / 255.0
```

Εικόνα 4.3 – Προετοιμασία Δεδομένων

4.5 Υλοποίηση Μοντέλων

4.5.1 Υλοποίηση Μοντέλου UNet

Με σκοπό την εξαγωγή των προβλεπόμενων μασκών για τα τρία χαρακτηριστικά ICM, TE, ZP των βλαστοκύστεων, υλοποιήθηκε ένα μοντέλο UNet με τρεις εξόδους. Το συγκεκριμένο μοντέλο αποτελείται από 34 επίπεδα. Πιο συγκεκριμένα, 1 επίπεδο εισόδου, 12 επίπεδα στο contracting path, 2 επίπεδα στο bottleneck, 16 επίπεδα στο expanding path και 3 επίπεδα εξόδου. Στο επίπεδο εισόδου, το μοντέλο δέχεται τις εικόνες σε διαστάσεις 128x128x3, δηλαδή με τρια κανάλια χρώματος (RGB).

Το contracting path αποτελείται από 4 διαδοχικά στάδια. Κάθε στάδιο περιλαμβάνει 2 convolutional layers και 1 max pooling layer, με σκοπό την μείωση των διαστάσεων των εικόνων εισόδου και την εξαγωγή των χαρακτηριστικών τους. Κάθε convolutional layer δημιουργεί φίλτρα 3x3, χρησιμοποιεί ReLU ως συνάρτηση ενεργοποίησης και διατηρεί τις αρχικές διαστάσεις των εικόνων (padding='same'). Πρέπει να σημειωθεί ότι, σε κάθε στάδιο, τα φίλτρα αυξάνονται (από 64 σε 128, από 128 σε 256 και από 256 σε 512), με σκοπό την εξαγωγή όλο και πιο περίπλοκων χαρακτηριστικών. Επιπλέον, χρησιμοποιήθηκαν max pooling layers με βηματισμό 2, για την μείωση των διαστάσεων των εικόνων. Αμέσως μετά, στο bottleneck, χρησιμοποιήθηκαν 2 διαδοχικά convolutional layers με 1024 φίλτρα 3x3, ReLU ως συνάρτηση ενεργοποίησης και padding='same' για την διατήρηση των διαστάσεων των εικόνων.

Το expanding path είναι υπεύθυνο για την αύξηση των διαστάσεων της εικόνας και την επίτευξη της τμηματοποίησης, μέσω των χαρακτηριστικών που έχουν εξαχθεί από τα επίπεδα του contracting path. Το expanding path αποτελείται από 4 διαδοχικά στάδια. Κάθε στάδιο περιλαμβάνει 1 upsampling layer, 1 skip connection (concatenate) και 2 convolutional layers. Κάθε upsampling layer με βηματισμό 2, χρησιμοποιήθηκε για την αύξηση των διαστάσεων των εικόνων. Αμέσως μετά, πραγματοποιήθηκε ο συνδυασμός του upsampling layer με τα αντίστοιχα επίπεδα του contracting path. Τέλος χρησιμοποιήθηκαν convolutional layers, τα οποία δημιουργούν φίλτρα 3x3, χρησιμοποιούν ReLU ως συνάρτηση ενεργοποίησης και διατηρούν τις αρχικές διαστάσεις των εικόνων (padding='same'). Πρέπει να σημειωθεί ότι, σε κάθε στάδιο, τα φίλτρα μειώνονται (από 512 σε 256, από 256 σε 128 και από 128 σε 64), με σκοπό την ανακατασκευή της εικόνας και την δημιουργία της μάσκας τμηματοποίησης.

Κάθε επίπεδο εξόδου παράγει ένα χάρτη τμηματοποίησης για κάθε χαρακτηριστικό ICM, TE, ZP των βλαστοκύστεων. Χρησιμοποιήθηκαν 3 convolutional layers, τα οποία δημιουργούν φίλτρα 1x1, με σκοπό την μείωση των φίλτρων και την εξαγωγή της μάσκας. Επιπλέον, χρησιμοποιήθηκε sigmoid ως συνάρτηση ενεργοποίησης, καθώς πρόκειται για περίπτωση δυαδικής τμηματοποίησης, δηλαδή την μετατροπή των εξόδων του μοντέλου σε 0 και 1.

Η κατασκευή του μοντέλου πραγματοποιήθηκε με τον ορισμό των τριών εξόδων, αλλά και των παραμέτρων του. Πιο αναλυτικά, χρησιμοποιήθηκε ο Adam ως βελτιστοποιητής με learning rate ίσο με 0.001, συνδυασμός binary cross-entropy και dice loss ως συνάρτηση κόστους, και Intersection of Union ως μετρική αξιολόγησης του μοντέλου. Η χρήση των δύο συναρτήσεων κόστους οδηγεί σε καλύτερη απόδοση για το

μοντέλο, καθώς συνδυάζει τα πλεονεκτήματα και από τις δύο. Στις εικόνες 4.4 και 4.5 παρουσιάζεται ο κώδικας για την υλοποίηση του μοντέλου UNet.

```python
# Ορισμός Dice Loss
def dice_loss(y_true, y_pred):
    smooth = 1e-6  # Για αποφυγή διαίρεσης με το μηδέν
    intersection = K.sum(K.abs(y_true * y_pred), axis=(1, 2, 3))
    union = K.sum(y_true, axis=(1, 2, 3)) + K.sum(y_pred, axis=(1, 2, 3))
    dice = (2. * intersection + smooth) / (union + smooth)
    return 1 - dice

# Combo Loss (Συνσυασμός Binary Crossentropy και Dice Loss)
def combo_loss(y_true, y_pred):
    bce = losses.BinaryCrossentropy()(y_true, y_pred)
    dice = dice_loss(y_true, y_pred)
    return bce + dice

# Κατασκευή U-Net μοντέλου με 3 εξόδους
def unet_model(input_size=(128, 128, 3)):
    inputs = Input(input_size)

    # Downsampling
    conv1 = layers.Conv2D(64, 3, activation='relu', padding='same')(inputs)
    conv1 = layers.Conv2D(64, 3, activation='relu', padding='same')(conv1)
    pool1 = layers.MaxPooling2D(pool_size=(2, 2))(conv1)

    conv2 = layers.Conv2D(128, 3, activation='relu', padding='same')(pool1)
    conv2 = layers.Conv2D(128, 3, activation='relu', padding='same')(conv2)
    pool2 = layers.MaxPooling2D(pool_size=(2, 2))(conv2)

    conv3 = layers.Conv2D(256, 3, activation='relu', padding='same')(pool2)
    conv3 = layers.Conv2D(256, 3, activation='relu', padding='same')(conv3)
    pool3 = layers.MaxPooling2D(pool_size=(2, 2))(conv3)

    conv4 = layers.Conv2D(512, 3, activation='relu', padding='same')(pool3)
    conv4 = layers.Conv2D(512, 3, activation='relu', padding='same')(conv4)
    pool4 = layers.MaxPooling2D(pool_size=(2, 2))(conv4)

    # Bottleneck
    conv5 = layers.Conv2D(1024, 3, activation='relu', padding='same')(pool4)
    conv5 = layers.Conv2D(1024, 3, activation='relu', padding='same')(conv5)
```

Εικόνα 4.4 – Υλοποίηση μοντέλου UNet 1/2

```
    # Upsampling
    up6 = layers.UpSampling2D(size=(2, 2))(conv5)
    merge6 = layers.concatenate([conv4, up6], axis=3)
    conv6 = layers.Conv2D(512, 3, activation='relu', padding='same')(merge6)
    conv6 = layers.Conv2D(512, 3, activation='relu', padding='same')(conv6)

    up7 = layers.UpSampling2D(size=(2, 2))(conv6)
    merge7 = layers.concatenate([conv3, up7], axis=3)
    conv7 = layers.Conv2D(256, 3, activation='relu', padding='same')(merge7)
    conv7 = layers.Conv2D(256, 3, activation='relu', padding='same')(conv7)

    up8 = layers.UpSampling2D(size=(2, 2))(conv7)
    merge8 = layers.concatenate([conv2, up8], axis=3)
    conv8 = layers.Conv2D(128, 3, activation='relu', padding='same')(merge8)
    conv8 = layers.Conv2D(128, 3, activation='relu', padding='same')(conv8)

    up9 = layers.UpSampling2D(size=(2, 2))(conv8)
    merge9 = layers.concatenate([conv1, up9], axis=3)
    conv9 = layers.Conv2D(64, 3, activation='relu', padding='same')(merge9)
    conv9 = layers.Conv2D(64, 3, activation='relu', padding='same')(conv9)

    # Έξοδοι για κάθε μάσκα (ICM, TE, ZP)
    output_icm = layers.Conv2D(1, 1, activation='sigmoid', name='icm_output')(conv9)
    output_te = layers.Conv2D(1, 1, activation='sigmoid', name='te_output')(conv9)
    output_zp = layers.Conv2D(1, 1, activation='sigmoid', name='zp_output')(conv9)

    model = Model(inputs=[inputs], outputs=[output_icm, output_te, output_zp])
    model.compile(
        optimizer=Adam(learning_rate=0.001),
        loss=combo_loss,
        metrics={
        'icm_output': [iou_metric],
        'te_output': [iou_metric],
        'zp_output': [iou_metric]}
    )
    return model

model = unet_model()
model.summary()  # Εμφάνιση της περίληψης του μοντέλου με τον αριθμό των παραμέτρων
```

Εικόνα 4.5 – Υλοποίηση μοντέλου UNet 2/2

4.5.2 Υλοποίηση Μοντέλου DeepLab

Το δεύτερο μοντέλο που υλοποιήθηκε με σκοπό την εξαγωγή των προβλεπόμενων μασκών για τα τρια χαρακτηριστικά ICM, TE, ZP των βλαστοκύστεων, είναι ένα μοντέλο DeepLabV3 τριών εξόδων, με χρήση του MobileNetV2 ως backbone. Επιπλέον, για την καταγραφή πληροφοριών σε διαφορετικές κλίμακες, το εν λόγω μοντέλο χρησιμοποιεί Atrous Spatial Pyramid Pooling (ASPP), το οποίο εκτελεί παράλληλα πολλαπλές atrous convolutions με διαφορετικούς ρυθμούς.

Όσον αφορά το ASPP, δημιουργήθηκε μια συνάρτηση, η οποία δέχεται σαν ορίσματα έναν χάρτη χαρακτηριστικών από το backbone, τους διαφορετικούς ρυθμούς (atrous_rates) και τα κανάλια (output_channels). Η έξοδος της συγκεκριμένης

συνάρτησης είναι ένας χάρτης χαρακτηριστικών, ο οποίος εισάγεται στο τμήμα του αποκωδικοποιητή (decoder). Αποτελείται από 1 convolution layer, το οποίο χρησιμοποιείται για την εκμάθηση χαρακτηριστικών. Το εν λόγω επίπεδο δημιουργεί φίλτρα με μέγεθος πυρήνα ίσο με 1, χρησιμοποιεί ReLU ως συνάρτηση ενεργοποίησης και διατηρεί τις διαστάσεις των εικόνων (padding='same'). Πρέπει να σημειωθεί πως ο αριθμός των φίλτρων που δημιουργεί το επίπεδο αυτό ορίζεται από το τρίτο όρισμα της συνάρτησης (output_channels). Στην συνέχεια, πραγματοποιείται η εκτέλεση πολλαπλών atrous convolutions με διαφορετικούς ρυθμούς (atrous rates), με σκοπό την καταγραφή πληροφοριών σε διαφορετικές κλίμακες. Οι ρυθμοί αυτοί επίσης δίνονται ως δεύτερο όρισμα στην συνάρτηση. Ακολουθεί μια διαδικασία, η οποία εκτελεί Global Average Pooling για την καταγραφή πληροφοριών από ολόκληρη την εικόνα. Τέλος, πραγματοποιείται η συγχώνευση των αποτελεσμάτων των παραπάνω διαδικασιών και η μείωση των διαστάσεων μέσω ενός convolutional layer, εξάγοντας τον τελικό χάρτη χαρακτηριστικών.

Όσον αφορά την υλοποίηση του μοντέλου DeepLabV3, επιλέχθηκε ως backbone το MobileNetV2, καθώς είναι ταχύτερο, έχει πιο μικρό αριθμό παραμέτρων και καταναλώνει λιγότερη μνήμη GPU σε σχέση με άλλα προεκπαιδευμένα μοντέλα. Το MobileNetV2, το οποίο έχει προεκπαιδευτεί σε ένα μεγάλο σύνολο δεδομένων (ImageNet), εξάγει τον χάρτη χαρακτηριστικών από το block_13_expand_relu. Αμέσως μετά, ακολουθεί η συνάρτηση για το ASPP, η οποία δέχεται σαν εισόδους την έξοδο του MobileNetV2, τους διαφορετικούς ρυθμούς (6, 12 και 18) και τα κανάλια (output_channels). Τα κανάλια ορίστηκαν σε 256, με σκοπό την μείωση της διάστασης, αλλά ταυτόχρονα και την διατήρηση της πληροφορίας. Στο τμήμα του αποκωδικοποιητή (decoder) χρησιμοποιήθηκαν 1 upsampling layer και 1 convolutional layer για την αύξηση των διαστάσεων του χάρτη χαρακτηριστικών και την εξαγωγή νέων χαρακτηριστικών αντίστοιχα. Το upsampling layer χρησιμοποιεί βηματισμό 4, ενώ το convolutional layer, δημιουργεί φίλτρα 3x3 και χρησιμοποιεί ReLU ως συνάρτηση ενεργοποίησης.

Με σκοπό την εξαγωγή του χάρτη τμηματοποίησης για κάθε χαρακτηριστικό ICM, TE, ZP των βλαστοκύστεων, δημιουργήθηκαν 3 επίπεδα εξόδου. Κάθε επίπεδο εξόδου χρησιμοποιεί 1 upsampling layer και 1 convolutional layer, με σκοπό την αύξηση της ανάλυσης και την εξαγωγή των προβλεπόμενων μασκών αντίστοιχα. Τα upsampling layers χρησιμοποιούν βηματισμό 4 και interpolation='bilinear' για εξομάλυνση των χαρακτηριστικών. Τα convolutional layers δημιουργούν φίλτρα 1x1 και χρησιμοποιούν sigmoid ως συνάρτηση ενεργοποίησης, καθώς πρόκειται για περίπτωση δυαδικής τμηματοποίησης, δηλαδή την μετατροπή των εξόδων του μοντέλου σε 0 και 1.

Όπως και στην περίπτωση του UNet, η κατασκευή του μοντέλου DeepLabV3 πραγματοποιήθηκε με τον ορισμό των τριών εξόδων, αλλά και των παραμέτρων του. Πιο αναλυτικά, χρησιμοποιήθηκε ο Adam ως βελτιστοποιητής με learning rate ίσο με 0.001, επιλέχθηκε συνδυασμός binary cross-entropy και dice loss ως συνάρτηση κόστους και χρησιμοποιήθηκε το Intersection of Union ως μετρική αξιολόγησης του μοντέλου. Στις Εικόνες 4.6 και 4.7 παρουσιάζεται ο κώδικας για την υλοποίηση του μοντέλου DeepLabV3 με χρήση MobileNetV2 ως backbone.

```python
# Ορισμός Dice Loss
def dice_loss(y_true, y_pred):
    smooth = 1e-6  # Για αποφυγή διαίρεσης με το μηδέν
    intersection = K.sum(K.abs(y_true * y_pred), axis=(1, 2, 3))
    union = K.sum(y_true, axis=(1, 2, 3)) + K.sum(y_pred, axis=(1, 2, 3))
    dice = (2. * intersection + smooth) / (union + smooth)
    return 1 - dice

# Συνδυασμένο Combo Loss (π.χ. Binary Crossentropy + Dice Loss)
def combo_loss(y_true, y_pred):
    bce = losses.BinaryCrossentropy()(y_true, y_pred)
    dice = dice_loss(y_true, y_pred)
    return bce + dice

def AtrousSpatialPyramidPooling(inputs, atrous_rates, output_channels):
    """Builds the ASPP module."""
    # 1x1 Convolution
    conv1x1 = layers.Conv2D(output_channels, kernel_size=1, padding='same', activation='relu')(inputs)

    # Atrous Convolutions
    atrous_convs = []
    for rate in atrous_rates:
        atrous_conv = layers.Conv2D(output_channels, kernel_size=3, padding='same',
                                    dilation_rate=rate, activation='relu')(inputs)
        atrous_convs.append(atrous_conv)

    # Image Pooling
    image_pooling = layers.GlobalAveragePooling2D()(inputs)
    image_pooling = layers.Reshape((1, 1, inputs.shape[-1]))(image_pooling)
    image_pooling = layers.Conv2D(output_channels, kernel_size=1, padding='same', activation='relu')(image_pooling)
    image_pooling = layers.UpSampling2D(size=(inputs.shape[1], inputs.shape[2]), interpolation='bilinear')(image_pooling)

    # Concatenate all outputs
    aspp_outputs = layers.Concatenate()([conv1x1] + atrous_convs + [image_pooling])

    # Final 1x1 Convolution
    aspp_outputs = layers.Conv2D(output_channels, kernel_size=1, padding='same', activation='relu')(aspp_outputs)

    return aspp_outputs
```

Εικόνα 4.6 – Υλοποίηση μοντέλου DeepLab με backbone το MobileNetV2 1/2

```python
def DeepLabV3(input_shape=(128, 128, 3), num_classes=1, backbone='mobilenetv2', atrous_rates=(6, 12, 18)):
    """Constructs the DeepLabV3 model."""
    base_model = tf.keras.applications.MobileNetV2(input_shape=input_shape, include_top=False, weights='imagenet')
    backbone_output = base_model.get_layer('block_13_expand_relu').output

    # ASPP Module
    aspp = AtrousSpatialPyramidPooling(backbone_output, atrous_rates, output_channels=256)

    # Decoder
    decoder_input = layers.UpSampling2D(size=(4, 4), interpolation='bilinear')(aspp)
    decoder_output = layers.Conv2D(256, kernel_size=3, padding='same', activation='relu')(decoder_input)

    output_icm = layers.UpSampling2D(size=(4, 4), interpolation='bilinear', name='upsampled_icm')(decoder_output)
    output_icm = layers.Conv2D(1, kernel_size=1, activation='sigmoid', name='icm_output')(output_icm)

    output_te = layers.UpSampling2D(size=(4, 4), interpolation='bilinear', name='upsampled_te')(decoder_output)
    output_te = layers.Conv2D(1, kernel_size=1, activation='sigmoid', name='te_output')(output_te)

    output_zp = layers.UpSampling2D(size=(4, 4), interpolation='bilinear', name='upsampled_zp')(decoder_output)
    output_zp = layers.Conv2D(1, kernel_size=1, activation='sigmoid', name='zp_output')(output_zp)

    model = Model(inputs=base_model.input, outputs=[output_icm, output_te, output_zp])
    model.compile(
        optimizer=Adam(learning_rate=0.001),
        loss=combo_loss,
        metrics={
            'icm_output': [iou_metric],
            'te_output': [iou_metric],
            'zp_output': [iou_metric]
        }
    )
    return model

# Δημιουργία του μοντέλου DeepLab με MobileNetV2
model = DeepLabV3(input_shape=(128, 128, 3), num_classes=1, backbone='mobilenetv2')
model.summary()  # Εμφάνιση της περίληψης του μοντέλου με τον αριθμό των παραμέτρων
```

Εικόνα 4.7 – Υλοποίηση μοντέλου DeepLab με backbone το MobileNetV2 2/2

4.5.3 Υλοποίηση Μοντέλου PSPNet

Το τρίτο μοντέλο που υλοποιήθηκε με σκοπό την εξαγωγή των προβλεπόμενων μασκών για τα τρια χαρακτηριστικά ICM, TE, ZP των βλαστοκύστεων, είναι ένα μοντέλο PSPNet τριών εξόδων, με χρήση του MobileNetV2 ως backbone. Επιπλέον, το εν λόγω μοντέλο χρησιμοποιεί Pyramid Pooling Module (PPM) για την καταγραφή τόσο τοπικών όσο και παγκόσμιων πληροφοριών σε διαφορετικές κλίμακες.

Όσον αφορά το PPM, δημιουργήθηκε μια συνάρτηση, η οποία δέχεται σαν όρισμα έναν χάρτη χαρακτηριστικών από το backbone. Η έξοδος της συγκεκριμένης συνάρτησης είναι ένας χάρτης χαρακτηριστικών, ο οποίος εισάγεται στο τμήμα του αποκωδικοποιητή (decoder). Αρχικά, πραγματοποιείται ο ορισμός των τριών επιπέδων pooling (2, 4 και 6) για την δημιουργία χαρακτηριστικών σε διαφορετικές κλίμακες. Πρέπει να σημειωθεί ότι τα επόμενα βήματα εφαρμόστηκαν για κάθε pool size. Εφαρμόστηκε ένα average pooling, με σκοπό την συμπίεση του χάρτη χαρακτηριστικών. Στην συνέχεια, χρησιμοποιήθηκε ένα convolutional layer για την μείωση της διάστασης των καναλιών. Αμέσως μετά, χρησιμοποιήθηκε batch normalization, με σκοπό την πιο αποδοτική εκμάθηση από το μοντέλο και την σταθερή κατανομή των δεδομένων στα

επίπεδα του. Επιπλέον, πραγματοποιείται η επαναφορά των διαστάσεων του χάρτη χαρακτηριστικών στο αρχικό μέγεθος. Για την συγκεκριμένη διαδικασία χρησιμοποιήθηκε η μέθοδος bilinear interpolation, καθώς οδηγεί σε πιο ομαλές μεταβάσεις. Ακολουθεί μια διαδικασία, η οποία εκτελεί Global Average Pooling για τον υπολογισμό του μέσου όρου ολόκληρου του χάρτη χαρακτηριστικών και την επαναφορά του στο αρχικό μέγεθος. Τέλος, πραγματοποιείται η συγχώνευση όλων των χαρακτηριστικών, τα οποία προέκυψαν από το Global Average Pooling και το pooling με 2x2, 4x4 και 6x6, με τον χάρτη χαρακτηριστικών από το backbone.

Όσον αφορά την υλοποίηση του μοντέλου PSPNet, επιλέχθηκε ως backbone το MobileNetV2, για τους λόγους που αναφέρονται στην υπόενοτητα 4.4.2. Το MobileNetV2 εξάγει τον χάρτη χαρακτηριστικών από το block_13_expand_relu. Αμέσως μετά, ακολουθεί η συνάρτηση για το PPM, η οποία δέχεται σαν είσοδο την έξοδο του MobileNetV2. Στο τμήμα του αποκωδικοποιητή (decoder) χρησιμοποιήθηκε 1 convolutional layer, το οποίο δημιουργεί 256 φίλτρα 3x3, με σκοπό την εκμάθηση περισσότερων χαρακτηριστικών. Χρησιμοποιήθηκε batch normalization, με σκοπό ταχύτερη σύγκλιση και πιο σταθερή εκπαίδευση. Επιπλέον, για την εισαγωγή μη-γραμμικότητας, χρησιμοποιήθηκε ReLU ως συνάρτηση ενεργοποίησης. Στην συνέχεια, προστέθηκε 1 upsampling layer 2x2 με bilinear interpolation, με σκοπό την ομαλή αύξηση των διαστάσεων των εικόνων. Η διαδικασία που μόλις προαναφέρθηκε για το τμήμα του κωδικοποιητή επαναλαμβάνεται ακόμη δύο φορές, με σκοπό την λεπτομερέστερη εξαγωγή χαρακτηριστικών και την επαναφορά της εικόνας στο αρχικό της μέγεθος. Πρέπει να σημειωθεί ότι, για την εξαγωγή λεπτομερέστερων πληροφοριών απαιτείται η σταδιακή μείωση των φίλτρων στα convolutional layers (από 256 σε 128 και από 128 σε 64).

Με σκοπό την εξαγωγή του χάρτη τμηματοποίησης για κάθε χαρακτηριστικό ICM, TE, ZP των βλαστοκύστεων, δημιουργήθηκαν 3 επίπεδα εξόδου. Κάθε επίπεδο εξόδου χρησιμοποιεί 1 upsampling layer και 1 convolutional layer, με σκοπό την αύξηση της ανάλυσης και την εξαγωγή των προβλεπόμενων μασκών αντίστοιχα. Τα upsampling layers χρησιμοποιούν βηματισμό 4 και interpolation='bilinear', για εξομάλυνση των χαρακτηριστικών. Τα convolutional layers δημιουργούν φίλτρα 1x1 και χρησιμοποιούν sigmoid ως συνάρτηση ενεργοποίησης, καθώς πρόκειται για περίπτωση δυαδικής τμηματοποίησης, δηλαδή την μετατροπή των εξόδων του μοντέλου σε 0 και 1.

Η κατασκευή του μοντέλου PSPNet πραγματοποιήθηκε με τον ορισμό των τριών εξόδων, αλλά και των παραμέτρων του. Πιο αναλυτικά, χρησιμοποιήθηκε ο Adam ως βελτιστοποιητής με learning rate ίσο με 0.001, επιλέχθηκε συνδυασμός binary cross-

entropy και dice loss ως συνάρτηση κόστους και χρησιμοποιήθηκε το Intersection of Union ως μετρική αξιολόγησης του μοντέλου. Στις Εικόνες 4.8 και 4.9 παρουσιάζεται ο κώδικας για την υλοποίηση του μοντέλου PSPNet, με χρήση MobileNetV2 ως backbone.

```python
# Συνάρτηση Pyramid Pooling Module
def pyramid_pooling_module(input_tensor):
    pool_sizes = [2, 4, 6]
    input_shape = tf.shape(input_tensor)
    h, w = input_shape[1], input_shape[2]

    # Λίστα για αποθήκευση των αποτελεσμάτων
    pooled_outputs = []

    # Shift Pooling
    for pool_size in pool_sizes:
        pooled = layers.AveragePooling2D(pool_size=pool_size, strides=pool_size, padding='same')(input_tensor)
        upsampled = layers.Conv2D(filters=input_tensor.shape[-1], kernel_size=1, padding='same', use_bias=False)(pooled)
        upsampled = layers.BatchNormalization()(upsampled)
        upsampled = tf.image.resize(upsampled, (h, w), method='bilinear')
        pooled_outputs.append(upsampled)

    # Global Average Pooling
    global_pooled = layers.GlobalAveragePooling2D()(input_tensor)
    global_pooled = layers.Reshape((1, 1, input_tensor.shape[-1]))(global_pooled)
    global_pooled = layers.Conv2D(filters=input_tensor.shape[-1], kernel_size=1, padding='same', use_bias=False)(global_pooled)
    global_pooled = layers.BatchNormalization()(global_pooled)
    global_pooled = tf.image.resize(global_pooled, (h, w), method='bilinear')

    pooled_outputs.append(global_pooled)

    # Συνένωση χαρακτηριστικών
    concatenated = layers.Concatenate()(pooled_outputs + [input_tensor])
    return concatenated
```

Εικόνα 4.8 – Υλοποίηση μοντέλου PSPNet με backbone το MobileNetV2 1/2

```python
# Συνάρτηση PSPNet
def pspnet_model(input_shape=(128, 128, 3), num_classes=1):
    # Φόρτωση του MobileNetV2 ως backbone
    base_model = tf.keras.applications.MobileNetV2(weights='imagenet', include_top=False, input_shape=input_shape)
    backbone_output = base_model.get_layer("block_13_expand_relu").output

    # Pyramid Pooling Module
    ppm_output = pyramid_pooling_module(backbone_output)

    # Decoder Layers
    x = layers.Conv2D(256, kernel_size=3, padding='same', use_bias=False)(ppm_output)
    x = layers.BatchNormalization()(x)
    x = layers.ReLU()(x)

    x = layers.UpSampling2D(size=(2, 2), interpolation='bilinear')(x)
    x = layers.Conv2D(128, kernel_size=3, padding='same', use_bias=False)(x)
    x = layers.BatchNormalization()(x)
    x = layers.ReLU()(x)

    x = layers.UpSampling2D(size=(2, 2), interpolation='bilinear')(x)
    x = layers.Conv2D(64, kernel_size=3, padding='same', use_bias=False)(x)
    x = layers.BatchNormalization()(x)
    x = layers.ReLU()(x)

    # Τρία επιμέρους outputs (ICM, TE, ZP)
    output_icm = layers.UpSampling2D(size=(4, 4), interpolation='bilinear')(x)
    output_icm = layers.Conv2D(1, 1, activation='sigmoid', name='icm_output')(output_icm)

    output_te = layers.UpSampling2D(size=(4, 4), interpolation='bilinear')(x)
    output_te = layers.Conv2D(1, 1, activation='sigmoid', name='te_output')(output_te)

    output_zp = layers.UpSampling2D(size=(4, 4), interpolation='bilinear')(x)
    output_zp = layers.Conv2D(1, 1, activation='sigmoid', name='zp_output')(output_zp)

    # Δημιουργία μοντέλου
    model = Model(inputs=base_model.input, outputs=[output_icm, output_te, output_zp])
    model.compile(
        optimizer=Adam(learning_rate=0.001),
        loss=combo_loss,
        metrics={
        'icm_output': [iou_metric],
        'te_output': [iou_metric],
        'zp_output': [iou_metric]}
    )
    return model

# Δημιουργία του μοντέλου PSPNet με MobileNetV2
input_shape = (128, 128, 3)
model = pspnet_model(input_shape, num_classes=1)
model.summary()  # Εμφάνιση της περίληψης του μοντέλου με τον αριθμό των παραμέτρων
```

Εικόνα 4.9 – Υλοποίηση μοντέλου PSPNet με backbone το MobileNetV2 2/2

4.6 Εκπαίδευση και Προβλέψεις των Μοντέλων

Τελικό στάδιο αποτελεί η εκπαίδευση του μοντέλου, η οποία είναι ύψιστης σημασίας, καθώς μέσα από την εκπαίδευση πραγματοποιείται η βελτιστοποίηση των βαρών του μοντέλου, με σκοπό την ακρίβεια των προβλέψεων. Πραγματοποιήθηκε ο διαχωρισμός των δεδομένων σε σύνολα εκπαίδευσης και δοκιμής, με αναλογία 80-20, δηλαδή 80% των εικόνων και των μασκών από ολόκληρο το σύνολο δεδομένων θα χρησιμοποιηθεί για την εκπαίδευση του μοντέλου και 20% θα χρησιμοποιηθεί για την δοκιμή του. Επίσης, ορίστηκε ο αριθμός των εποχών σε 20 και το μέγεθος των batches σε 8. Ο αριθμός των εποχών θέλει ιδιαίτερη προσοχή, καθώς ένας μικρός αριθμός μπορεί να οδηγήσει σε μη επαρκή εκπαίδευση του μοντέλου ενώ αντίθετα ένας μεγάλος αριθμός μπορεί να οδηγήσει σε υπερεκπαίδευση. Μετά την εκπαίδευση, ακολουθεί η διαδικασία κατά την οποία το μοντέλο κάνει προβλέψεις για τα δεδομένα εισόδου. Το αποτέλεσμα θα είναι μια μάσκα τμηματοποίησης για κάθε χαρακτηριστικό των βλαστοκύστεων. Στην Εικόνα 4.10 παρουσιάζεται ο κώδικας για την διαδικασία της εκπαίδευσης και των προβλέψεων για κάθε μοντέλο.

```
# Χωρισμός δεδομένων σε σύνολα εκπαίδευσης και δοκιμής
X_train, X_test, y_train_icm, y_test_icm = train_test_split(images, masks_icm, test_size=0.2, random_state=42)
_, _, y_train_te, y_test_te = train_test_split(images, masks_te, test_size=0.2, random_state=42)
_, _, y_train_zp, y_test_zp = train_test_split(images, masks_zp, test_size=0.2, random_state=42)

# Εκπαίδευση του μοντέλου
history = model.fit(X_train, [y_train_icm, y_train_te, y_train_zp], epochs=20, batch_size=8,
                    validation_data=(X_test, [y_test_icm, y_test_te, y_test_zp]))

# Προβλέψεις
predicted_masks_icm, predicted_masks_te, predicted_masks_zp = model.predict(X_test)
```

Εικόνα 4.10 – Εκπαίδευση & Προβέψεις των μοντέλων

4.7 Μέθοδοι Αξιολόγησης

Η απόδοση του μοντέλου αξιολογείται μετά από κάθε εποχή εκπαίδευσης και με αυτόν τον τρόπο βλέπουμε πως εξελίσσεται η απόδοσή του με το πέρασμα των εποχών, καθώς και αν το μοντέλο είναι πιθανό να καταλήξει σε υπερεκπαίδευση ή υποεκπαίδευση. Η μετρική αξιολόγησης που χρησιμοποιήθηκε σε αυτήν την έρευνα είναι το Intersection of Union. Στην Εικόνα 4.11 παρουσιάζεται ο κώδικας υπολογισμού της εν λόγω μετρικής.

```python
# Συνάρτηση υπολογισμού IoU
def iou_metric(y_true, y_pred):
    y_true = K.flatten(y_true)
    y_pred = K.flatten(y_pred)
    intersection = K.sum(y_true * y_pred)
    union = K.sum(y_true) + K.sum(y_pred) - intersection
    return intersection / (union + K.epsilon())
```

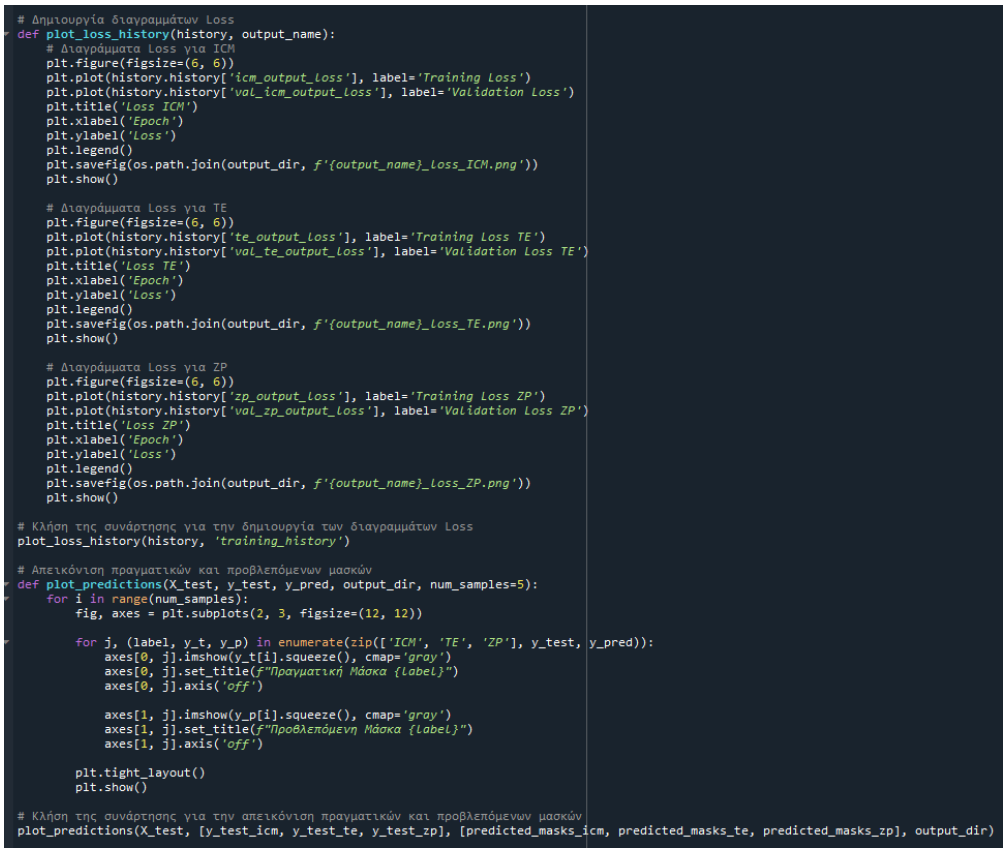
Εικόνα 4.11 – Μετρική Αξιολόγησης IoU

4.8 Απεικόνιση Αποτελεσμάτων

Αφού ολοκληρωθεί η διαδικασία της εκπαίδευσης, είναι απαραίτητο να αξιολογηθούν τα αποτελέσματα. Στις Εικόνες 4.12 και 4.13, παρουσιάζεται ο κώδικας για την απεικόνιση των αποτελεσμάτων, τα οποία παρατίθονται στο κεφάλαιο 5.

```python
# Εμφάνιση των πρώτων 10 εικόνων εισόδου και των αντίστοιχων προβλέψεων
for i in range(10):
    fig, axes = plt.subplots(1, 4, figsize=(20, 5))

    # Αρχική εικόνα εισόδου
    axes[0].imshow(X_test[i])
    axes[0].set_title("Εικόνα Εισόδου")
    axes[0].axis('off')

    # Προβλεπόμενες μάσκες
    axes[1].imshow(predicted_masks_icm[i].squeeze(), cmap='gray')
    axes[1].set_title("Προβλεπόμενη Μάσκα ICM")
    axes[1].axis('off')

    axes[2].imshow(predicted_masks_te[i].squeeze(), cmap='gray')
    axes[2].set_title("Προβλεπόμενη Μάσκα TE")
    axes[2].axis('off')

    axes[3].imshow(predicted_masks_zp[i].squeeze(), cmap='gray')
    axes[3].set_title("Προβλεπόμενη Μάσκα ZP")
    axes[3].axis('off')

    # Αποθήκευση των μασκών σε αρχεία
    cv2.imwrite(os.path.join(output_dir, f'pred_mask_icm_{i}.png'), predicted_masks_icm[i].squeeze() * 255)
    cv2.imwrite(os.path.join(output_dir, f'pred_mask_te_{i}.png'), predicted_masks_te[i].squeeze() * 255)
    cv2.imwrite(os.path.join(output_dir, f'pred_mask_zp_{i}.png'), predicted_masks_zp[i].squeeze() * 255)

    plt.show()

# Δημιουργία διαγραμμάτων IoU
def plot_iou_history(history, output_name):
    # Διαγράμματα IoU για ICM
    plt.figure(figsize=(6, 6))
    plt.plot(history.history['icm_output_iou_metric'], label='Training IoU ICM')
    plt.plot(history.history['val_icm_output_iou_metric'], label='Validation IoU ICM')
    plt.title('IoU ICM')
    plt.xlabel('Epoch')
    plt.ylabel('IoU')
    plt.legend()
    plt.savefig(os.path.join(output_dir, f'{output_name}_iou_ICM.png'))
    plt.show()

    # Διαγράμματα IoU για TE
    plt.figure(figsize=(6, 6))
    plt.plot(history.history['te_output_iou_metric'], label='Training IoU TE')
    plt.plot(history.history['val_te_output_iou_metric'], label='Validation IoU TE')
    plt.title('IoU TE')
    plt.xlabel('Epoch')
    plt.ylabel('IoU')
    plt.legend()
    plt.savefig(os.path.join(output_dir, f'{output_name}_iou_TE.png'))
    plt.show()

    # Διαγράμματα IoU για ZP
    plt.figure(figsize=(6, 6))
    plt.plot(history.history['zp_output_iou_metric'], label='Training IoU ZP')
    plt.plot(history.history['val_zp_output_iou_metric'], label='Validation IoU ZP')
    plt.title('IoU ZP')
    plt.xlabel('Epoch')
    plt.ylabel('IoU')
    plt.legend()
    plt.savefig(os.path.join(output_dir, f'{output_name}_iou_ZP.png'))
    plt.show()

# Κλήση της συνάρτησης για την δημιουργία των διαγραμμάτων IoU
plot_iou_history(history, 'training_history')
```

Εικόνα 4.12 – Απεικόνιση αποτελεσμάτων 1/2



```python
# Δημιουργία διαγραμμάτων Loss
def plot_loss_history(history, output_name):
    # Διαγράμματα Loss για ICM
    plt.figure(figsize=(6, 6))
    plt.plot(history.history['icm_output_loss'], label='Training Loss')
    plt.plot(history.history['val_icm_output_loss'], label='Validation Loss')
    plt.title('Loss ICM')
    plt.xlabel('Epoch')
    plt.ylabel('Loss')
    plt.legend()
    plt.savefig(os.path.join(output_dir, f'{output_name}_loss_ICM.png'))
    plt.show()

    # Διαγράμματα Loss για TE
    plt.figure(figsize=(6, 6))
    plt.plot(history.history['te_output_loss'], label='Training Loss TE')
    plt.plot(history.history['val_te_output_loss'], label='Validation Loss TE')
    plt.title('Loss TE')
    plt.xlabel('Epoch')
    plt.ylabel('Loss')
    plt.legend()
    plt.savefig(os.path.join(output_dir, f'{output_name}_loss_TE.png'))
    plt.show()

    # Διαγράμματα Loss για ZP
    plt.figure(figsize=(6, 6))
    plt.plot(history.history['zp_output_loss'], label='Training Loss ZP')
    plt.plot(history.history['val_zp_output_loss'], label='Validation Loss ZP')
    plt.title('Loss ZP')
    plt.xlabel('Epoch')
    plt.ylabel('Loss')
    plt.legend()
    plt.savefig(os.path.join(output_dir, f'{output_name}_loss_ZP.png'))
    plt.show()

# Κλήση της συνάρτησης για την δημιουργία των διαγραμμάτων Loss
plot_loss_history(history, 'training_history')

# Απεικόνιση πραγματικών και προβλεπόμενων μασκών
def plot_predictions(X_test, y_test, y_pred, output_dir, num_samples=5):
    for i in range(num_samples):
        fig, axes = plt.subplots(2, 3, figsize=(12, 12))

        for j, (label, y_t, y_p) in enumerate(zip(['ICM', 'TE', 'ZP'], y_test, y_pred)):
            axes[0, j].imshow(y_t[i].squeeze(), cmap='gray')
            axes[0, j].set_title(f"Πραγματική Μάσκα {label}")
            axes[0, j].axis('off')

            axes[1, j].imshow(y_p[i].squeeze(), cmap='gray')
            axes[1, j].set_title(f"Προβλεπόμενη Μάσκα {label}")
            axes[1, j].axis('off')

        plt.tight_layout()
        plt.show()

# Κλήση της συνάρτησης για την απεικόνιση πραγματικών και προβλεπόμενων μασκών
plot_predictions(X_test, [y_test_icm, y_test_te, y_test_zp], [predicted_masks_icm, predicted_masks_te, predicted_masks_zp], output_dir)
```

Εικόνα 4.13 – Απεικόνιση αποτελεσμάτων 2/2

Δημιουργήθηκαν τα διαγράμματα της μετρικής IoU, αλλά και της απώλειας, καθ' όλη την διάρκεια της εκπαίδευσης και της επικύρωσης των τριών μοντέλων, μετά από κάθε εποχή. Τα διαγράμματα αυτά μας δείχνουν το πως εξελίσσονται οι τιμές της μετρικής IoU και της απώλειας με το πέρασμα των εποχών, με σκοπό την αξιολόγηση των μοντέλων.

5. Αποτελέσματα

5.1 Εισαγωγή

Στο Κεφάλαιο 5 παρουσιάζονται τα αποτελέσματα που εξήχθησαν από την συγκεκριμένη έρευνα. Αρχικά, παρατίθενται τα αποτελέσματα για την εύρεση του κατάλληλου συνδυασμού των τιμών των υπερπαραμέτρων εκπαίδευσης (αριθμός εποχών, μέγεθος παρτίδας και ρυθμός εκμάθησης), με σκοπό την επίτευξη της μέγιστης δυνατής απόδοσης για τα τρία μοντέλα. Στην συνέχεια, παρουσιάζονται οι προβλεπόμενες μάσκες για τα τρια χαρακτηριστικά ICM, TE και ZP των βλαστοκύστεων, καθώς και οι τιμές της μετρικής IoU, που εξήχθησαν από τα τρία μοντέλα. Επίσης, παρατίθενται τα διαγράμματα τόσο της μετρικής αξιολόγησης IoU όσο και της απώλειας, καθ' όλη την διάρκεια της εκπαίδευσης και της επικύρωσης του μοντέλου, μετά από κάθε εποχή. Τέλος, πραγματοποιείται η σύγκριση των αποτελεσμάτων, καθώς και η ανάδειξη του μοντέλου με την καλύτερη επίδοση, δηλαδή με τις μεγαλύτερες τιμές για την μετρική αξιολόγησης IoU.

5.2 Αποτελέσματα για το μοντέλο UNet

Αρχικά, πραγματοποιήθηκε η διαδικασία εξαγωγής των προβλεπόμενων μασκών για διαφορετικούς συνδυασμούς των υπερπαραμέτρων εκπαίδευσης, με σκοπό την εύρεση των τιμών των υπερπαραμέτρων για τις οποίες το μοντέλο UNet παρουσιάζει την καλύτερη απόδοση. Πιο αναλυτικά, η τιμή των εποχών κυμαίνεται από 10 έως 20, η τιμή του learning rate από 0.0001 σε 0.001 και η τιμή του batch size από 8 σε 16, και από 16 σε 32. Στην συνέχεια, παρατίθεται ο Πίνακας 5.1, ο οποίος περιέχει τις τιμών της μετρικής αξιολόγησης IoU για κάθε πιθανό συνδυασμό των τιμών των υπερπαραμέτρων εκπαίδευσης.

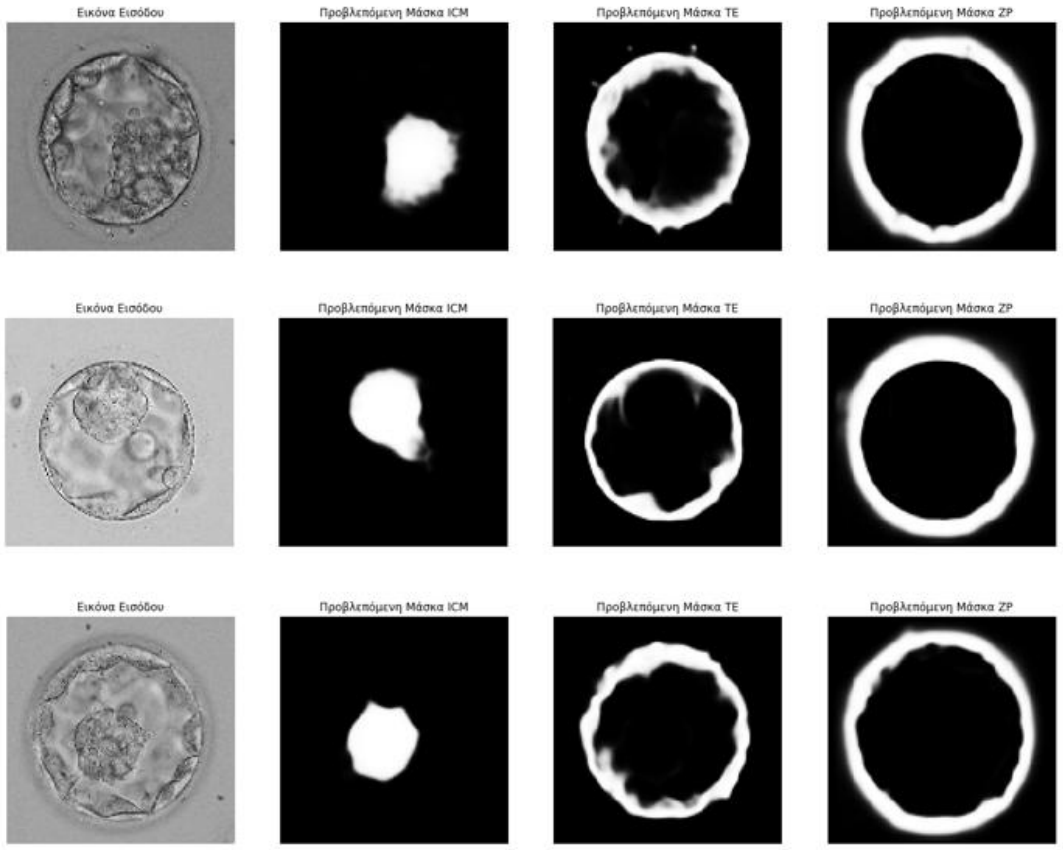
Πίνακας 5.1 – Αποτελέσματα των τιμών της μετρικής IoU για διαφορετικούς συνδυασμούς των υπερπαραμέτρων εκπαίδευσης του μοντέλου UNet

Παράμετροι Εκπαίδευσης	**Intersection of Union**		
	ICM	**TE**	**ZP**
Epochs = 10 Batch size = 8 Learning Rate = 0.0001	0.6333	0.4870	0.5588

Epochs = 10 Batch size = 8 Learning Rate = 0.001	0.6030	0.4913	0.5446
Epochs = 10 Batch size = 16 Learning Rate = 0.0001	0.5711	0.3929	0.5254
Epochs = 10 Batch size = 16 Learning Rate = 0.001	0.3658	0.5047	0.3069
Epochs = 10 Batch size = 32 Learning Rate = 0.0001	0.2295	0.2067	0.3122
Epochs = 10 Batch size = 32 Learning Rate = 0.001	0.1160	0.2301	0.1564
Epochs = 20 Batch size = 8 Learning Rate = 0.0001	0.6856	0.5494	0.6186
Epochs = 20 Batch size = 8 Learning Rate = 0.001	0.6705	0.5688	0.6365
Epochs = 20 Batch size = 16 Learning Rate = 0.0001	0.6112	0.5372	0.5452
Epochs = 20 Batch size = 16 Learning Rate = 0.001	0.6607	0.5543	0.6275
Epochs = 20 Batch size = 32 Learning Rate = 0.0001	0.5443	0.3679	0.5265
Epochs = 20 Batch size = 32 Learning Rate = 0.001	0.3577	0.3918	0.4813

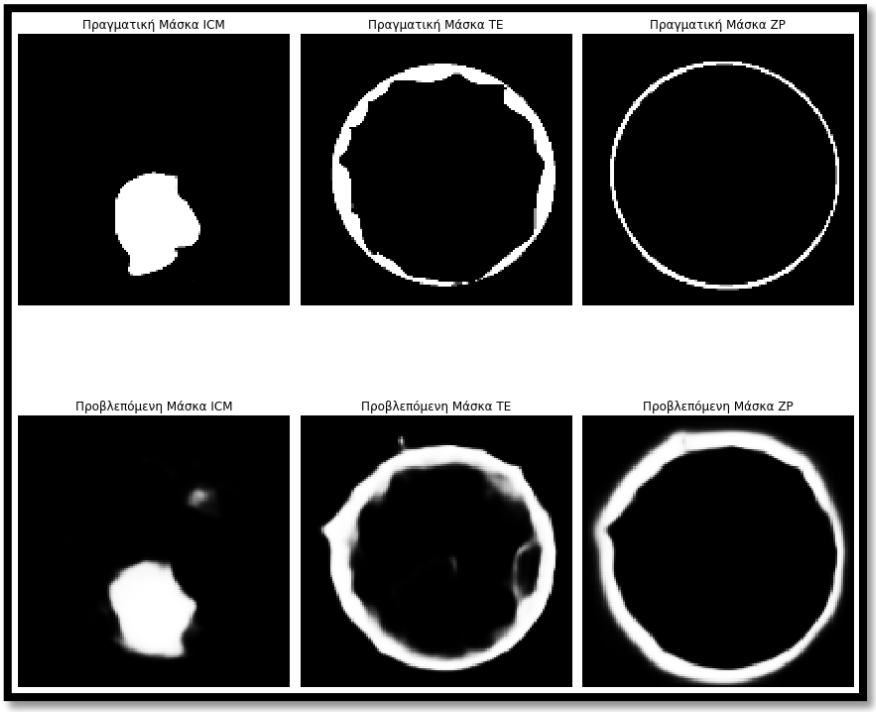
Μετά την παράθεση των αποτελεσμάτων του Πίνακα 5.1, γίνεται εύκολα αντιληπτό ότι το μοντέλο UNet παρουσιάζει την καλύτερη απόδοση για αριθμό εποχών ίσο με 20, τιμή του learning rate ίση με 0.001 και τιμή του batch size ίση με 8. Επομένως, ακολουθεί η παράθεση των εικόνων των προβλεπόμενων μασκών τμηματοποίησης για τα τρια χαρακτηριστικά ICM, TE και ZP, καθώς και των διαγραμμάτων της μετρικής αξιολόγησης IoU και της απώλειας κατά την διάρκεια των εποχών, για τον συγκεκριμένο συνδυασμό υπερπαραμέτρων. Στην Εικόνα 5.1, παρουσιάζεται ένα δείγμα από τις μάσκες που εξήχθησαν από το μοντέλο UNet. Πιο συγκεκριμένα, παρουσιάζεται ένα δείγμα από τις αρχικές εικόνες που δόθηκαν σαν είσοδος στο δίκτυο, καθώς και οι τρεις

μάσκες ICM, TE και ZP, οι οποίες είναι οι έξοδοι του δικτύου για κάθε εικόνα εισόδου αντίστοιχα.

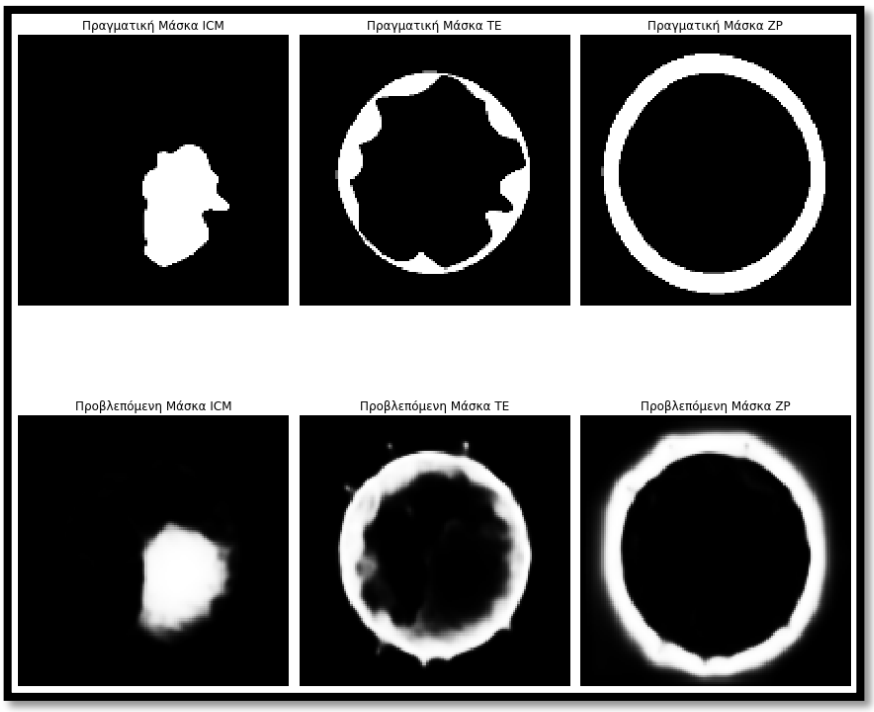


Εικόνα 5.1 – Προβλεπόμενες μάσκες ICM, TE & ZP για το μοντέλο UNet

Στις Εικόνες 5.2 και 5.3, παρουσιάζονται οι μάσκες ICM, TE, ZP που εξήχθησαν από το μοντέλο σε σύγκριση με τις πραγματικές μάσκες, για δύο τυχαίες εικόνες βλαστοκύστεων του συνόλου δεδομένων, με σκοπό την ανίχνευση των διαφορών τους.



Εικόνα 5.2 – Σύγκριση προβλεπόμενων και πραγματικών μασκών ICM, TE & ZP για το μοντέλο UNet 1/2



Εικόνα 5.3 – Σύγκριση προβλεπόμενων και πραγματικών μασκών ICM, TE & ZP για το μοντέλο UNet 2/2

Επίσης, στον Πίνακα 5.2, παρουσιάζονται τα συγκεντρωτικά αποτελέσματα που εξήχθησαν για την μετρική αξιολόγησης IoU, καθ' όλη την διάρκεια της εκπαίδευσης και της επικύρωσης του μοντέλου, μετά από κάθε εποχή ενώ στον Πίνακα 5.3, παρουσιάζονται τα συγκεντρωτικά αποτελέσματα που εξήχθησαν για την απώλεια, καθ' όλη την διάρκεια της εκπαίδευσης και της επικύρωσης του μοντέλου, μετά από κάθε εποχή.

Πίνακας 5.2 – Συγκεντρωτικά αποτελέσματα της μετρικής αξιολόγησης IoU καθ' όλη την διάρκεια της εκπαίδευσης και της επικύρωσης του μοντέλου UNet, μετά από κάθε εποχή

Εποχές	**Training IoU**			**Validation IoU**		
	ICM	**TE**	**ZP**	**ICM**	**TE**	**ZP**
1	0.0540	0.0787	0.1046	0.0563	0.0738	0.0940
2	0.0888	0.1172	0.1160	0.1149	0.1540	0.1287
3	0.1086	0.1355	0.1183	0.0526	0.0793	0.1131
4	0.0899	0.1199	0.1127	0.1234	0.1544	0.1413
5	0.1595	0.2001	0.1613	0.2254	0.2183	0.2143
6	0.2317	0.1986	0.2785	0.2594	0.2085	0.1644
7	0.3536	0.2799	0.3599	0.4794	0.3829	0.5017
8	0.4988	0.3960	0.4819	0.4137	0.4259	0.4754
9	0.4918	0.4239	0.4795	0.5044	0.4507	0.5299
10	0.5349	0.4475	0.5155	0.5759	0.4840	0.5217
11	0.5626	0.4539	0.5229	0.5456	0.4789	0.5355
12	0.5748	0.4637	0.5328	0.6166	0.5008	0.5838
13	0.5992	0.4782	0.5618	0.6110	0.4770	0.5511
14	0.6155	0.4898	0.5699	0.6402	0.5245	0.5745
15	0.6194	0.5087	0.5810	0.6435	0.5309	0.6052
16	0.6434	0.5141	0.5880	0.6511	0.5373	0.5875
17	0.6641	0.5363	0.6117	0.6687	0.5661	0.5843
18	0.6662	0.5480	0.6192	0.6577	0.5255	0.6056
19	0.6573	0.5461	0.6117	0.6833	0.5597	0.6089
20	0.6816	0.5627	0.6131	0.6705	0.5688	0.6365

Πίνακας 5.3 – Συγκεντρωτικά αποτελέσματα της απώλειας καθ' όλη την διάρκεια της εκπαίδευσης και της επικύρωσης του μοντέλου UNet, μετά από κάθε εποχή

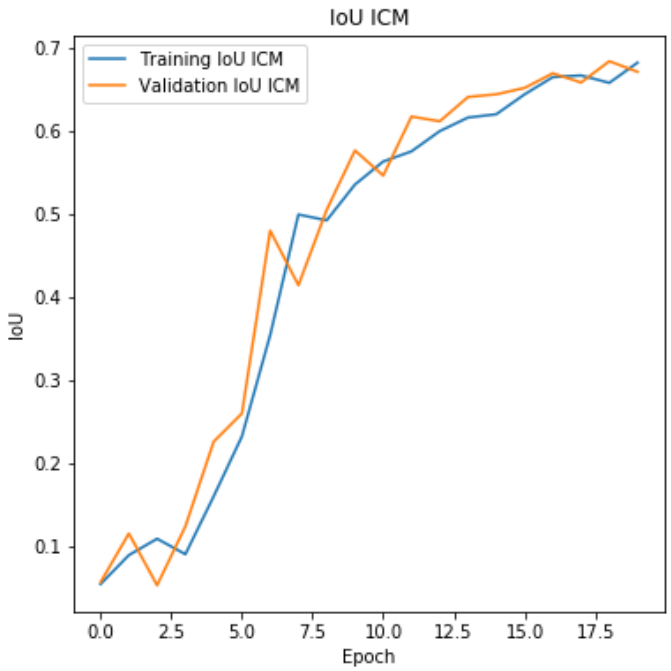
Εποχές	**Training Loss**			**Validation Loss**		
	ICM	**TE**	**ZP**	**ICM**	**TE**	**ZP**
1	12.8157	12.3001	7.9372	1.1769	1.2134	1.2867
2	1.0781	1.1053	1.2433	1.0382	1.0389	1.2075
3	1.0824	1.1670	1.3270	1.1640	1.1779	1.2788
4	1.0804	1.0960	1.2207	1.0380	1.0410	1.1760
5	0.9628	0.9387	1.1098	0.7972	0.8944	0.9986
6	0.8547	0.9704	0.9211	0.7244	0.9243	1.2535
7	0.6720	0.7887	0.7780	0.5388	0.6279	0.5874
8	0.5059	0.6377	0.5955	0.5553	0.6314	0.6202
9	0.5090	0.6141	0.6088	0.4881	0.5705	0.5190
10	0.4518	0.5717	0.5554	0.3724	0.5289	0.5446
11	0.4145	0.5757	0.5339	0.4144	0.5301	0.5540
12	0.3974	0.5526	0.5252	0.3446	0.4991	0.4782
13	0.3752	0.5454	0.4969	0.3309	0.5613	0.5134
14	0.3516	0.5176	0.4824	0.3245	0.4817	0.4944
15	0.3492	0.4999	0.4630	0.3165	0.4672	0.4483
16	0.3169	0.4931	0.4533	0.3017	0.4601	0.4572
17	0.2914	0.4643	0.4240	0.3091	0.4394	0.4659
18	0.2914	0.4547	0.4133	0.3060	0.4681	0.4224
19	0.3025	0.4524	0.4268	0.2745	0.4381	0.4555
20	0.2731	0.4338	0.4235	0.2678	0.4453	0.4098

Κάνοντας χρήση των παραπάνω αποτελεσμάτων, εξήχθησαν τα διαγράμματα τόσο της μετρικής αξιολόγησης IoU, όσο και της απώλειας, καθ' όλη την διάρκεια της εκπαίδευσης και της επικύρωσης του μοντέλου UNet, μετά από κάθε εποχή. Στις Εικόνες 5.4, 5.5 και 5.6, παρουσιάζονται τα διαγράμματα της μετρικής αξιολόγησης IoU, για κάθε προβλεπόμενη μάσκα ICM, TE, ZP.

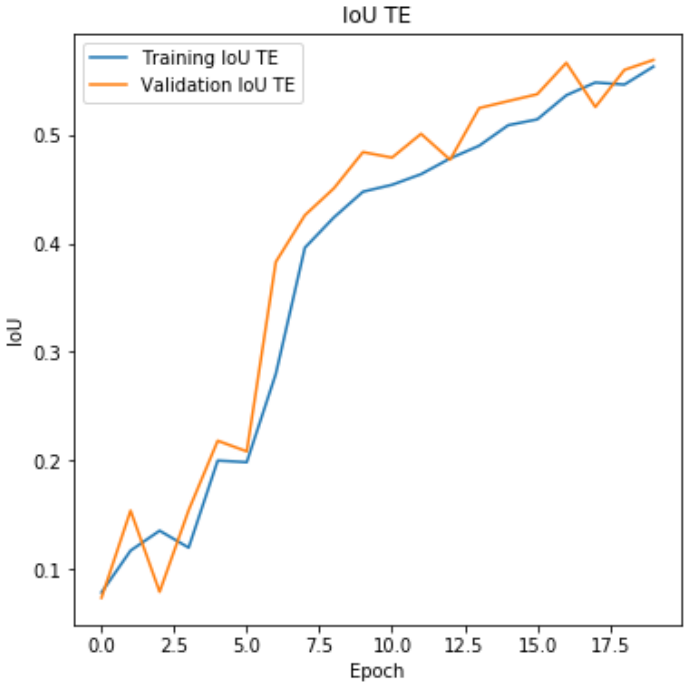
Είναι εμφανές ότι η τιμή της μετρικής αξιολόγησης IoU για την μάσκα ICM αυξάνεται σταθερά με το πέρασμα των εποχών, τόσο κατά την εκπαίδευση, όσο και κατά την επικύρωση του μοντέλου UNet. Επιπλέον, παρατηρείται ότι το εν λόγω μοντέλο

διαθέτει καλή γενίκευση, καθώς οι καμπύλες τόσο του Training IoU, όσο και του Validation IoU, ακολουθούν η μια την άλλη. Επομένως, το μοντέλο UNet εμφανίζει την ίδια απόδοση για τα δεδομένα εκπαίδευσης και επικύρωσης της μάσκας ICM, χωρίς να παρουσιάζεται το φαινόμενο του overfitting. Τέλος, πρέπει να σημειωθεί ότι επιτυγχάνεται αρκετά υψηλή απόδοση για την κατηγορία ICM, καθώς οι δύο καμπύλες φθάνουν πολύ κοντά στην τιμή 0.7, και πιο συγκεκριμένα, η καμπύλη του Training IoU φθάνει στην τιμή 0.6816 ενώ η καμπύλη του Validation IoU φθάνει στην τιμή 0.6705.

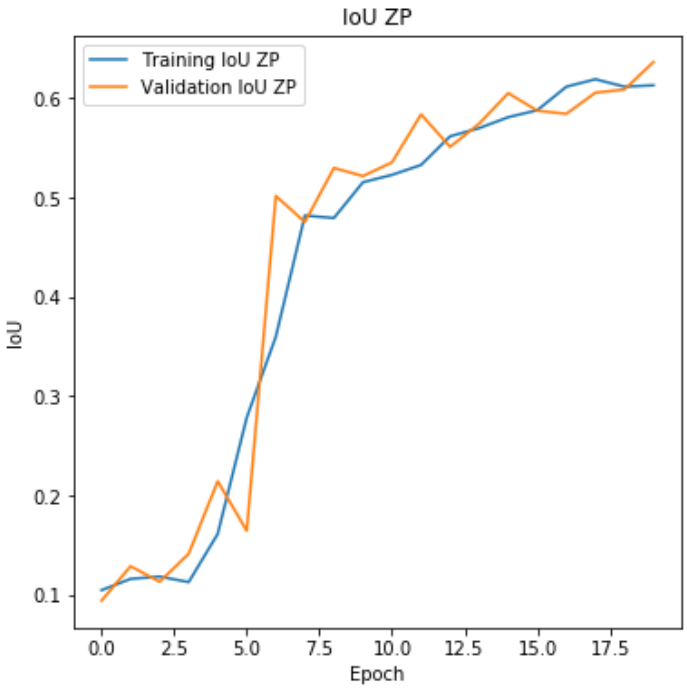
Το μοντέλο UNet εμφανίζει παρόμοια συμπεριφορά και για την μάσκα TE, καθώς η τιμές του Training IoU και του Validation IoU αυξάνονται σταθερά με το πέρασμα των εποχών. Και στην περίπτωση της μάσκας TE, οι δυο καμπύλες είναι αρκετά κοντά μεταξύ τους, το οποίο υποδηλώνει ότι πρόκειται για ένα μοντέλο με σταθερή και ισορροπημένη μάθηση. Τέλος, πρέπει να σημειωθεί ότι επιτυγχάνεται χαμηλότερη απόδοση για την κατηγορία TE, σε σχέση με την απόδοση της κατηγορίας ICM. Παρόλα αυτά, είναι αρκετά ικανοποιητική, καθώς οι δύο καμπύλες φθάνουν πολύ κοντά στην τιμή 0.6, και πιο συγκεκριμένα, η καμπύλη του Training IoU φθάνει στην τιμή 0.5627 ενώ η καμπύλη του Validation IoU φθάνει στην τιμή 0.5688.



Εικόνα 5.4 – Διάγραμμα μεταβολής του IoU σε κάθε εποχή για την προβλεπόμενη μάσκα ICM (UNet)



Εικόνα 5.5 – Διάγραμμα μεταβολής του IoU σε κάθε εποχή για την προβλεπόμενη μάσκα TE (UNet)



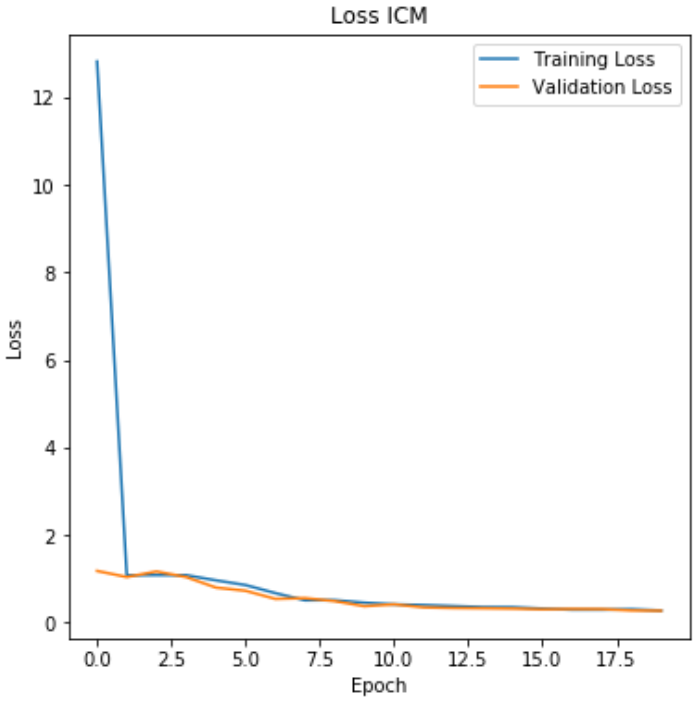
Εικόνα 5.6 – Διάγραμμα μεταβολής του IoU σε κάθε εποχή για την προβλεπόμενη μάσκα ZP (UNet)

Όσον αφορά την συμπεριφορά του μοντέλου UNet για την μάσκα ZP, παρατηρείται σταθερή αύξηση των καμπυλών, σε συνδυασμό με την εμφάνιση κάποιων μικρών αποκλίσεων μεταξύ τους. Παρόλα αυτά, η συμπεριφορά του μοντέλου για την μάσκα ZP είναι παρόμοια με αυτήν για την μάσκα TE. Τέλος, πρέπει να σημειωθεί ότι επιτυγχάνεται μια αρκετά ικανοποιητική απόδοση, καθώς οι δύο καμπύλες ξεπερνούν την τιμή 0.6, και πιο συγκεκριμένα, η καμπύλη του Training IoU φθάνει στην τιμή 0.6131 ενώ η καμπύλη του Validation IoU φθάνει στην τιμή 0.6365. Στις Εικόνες 5.7, 5.8 και 5.9, παρουσιάζονται τα διαγράμματα της απώλειας, για κάθε προβλεπόμενη μάσκα ICM, TE, ZP.

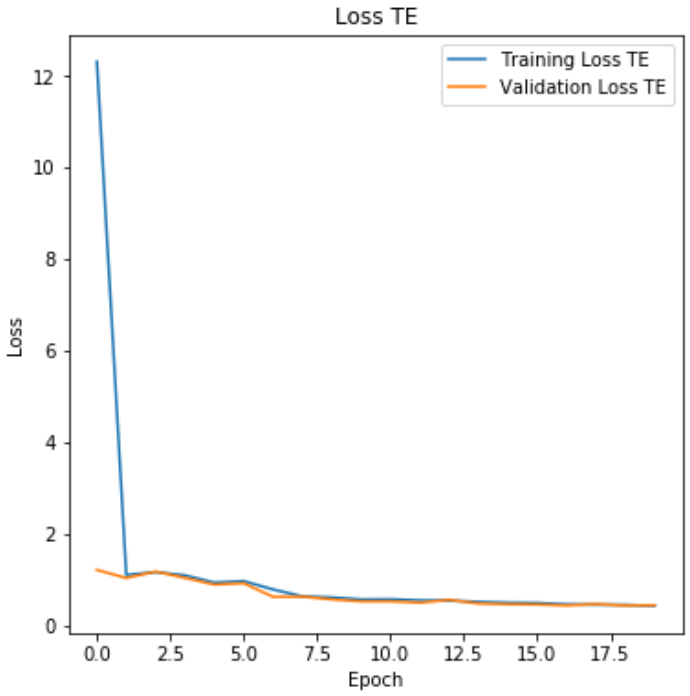
Στο διάγραμμα της απώλειας για την μάσκα ICM, είναι εμφανές ότι η καμπύλη του Training Loss ξεκινάει από μια υψηλή τιμή και πιο συγκεκριμένα την τιμή 12.8157, αλλά στην δεύτερη εποχή παρουσιάζει μια απότομη μείωση, με αποτέλεσμα οι δύο καμπύλες του Training Loss και του Validation Loss να ακολουθούν σχεδόν την ίδια πορεία, με ελάχιστες διαφορές. Το μοντέλο φθάνει στην τιμή 0.2731 για το Training Loss και στην τιμή 0.2678 για το Validation Loss. Όλα τα παραπάνω οδηγούν στο συμπέρασμα ότι πρόκειται για ένα μοντέλο με γρήγορη εκμάθηση και αποτελεσματική σύγκλιση.

Όσον αφορά την απώλεια για την μάσκα TE, είναι εμφανές ότι η καμπύλη του Training Loss ξεκινάει από μια υψηλή τιμή, και πιο συγκεκριμένα την τιμή 12.3001, αλλά στην δεύτερη εποχή παρουσιάζει μια απότομη μείωση, με αποτέλεσμα οι καμπύλες του Training Loss και του Validation Loss να ακολουθούν σχεδόν την ίδια πορεία, χωρίς απότομες διακυμάνσεις, το οποίο υποδηλώνει σταθερότητα στην εκπαίδευση. Το μοντέλο φθάνει στην τιμή 0.4338 για το Training Loss και στην τιμή 0.4453 για το Validation Loss.

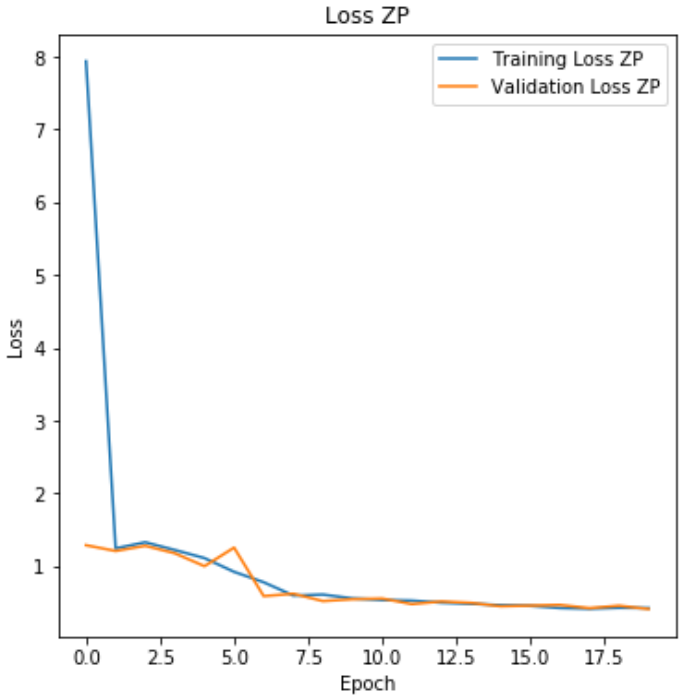
Στο διάγραμμα της απώλειας για την μάσκα ZP, είναι εμφανές ότι η καμπύλη του Training Loss ξεκινάει από μια υψηλή τιμή, και πιο συγκεκριμένα την τιμή 7.9372, αλλά στην δεύτερη εποχή παρουσιάζει μια απότομη μείωση, με αποτέλεσμα οι καμπύλες του Training Loss και του Validation Loss να ακολουθούν σχεδόν την ίδια πορεία. Πρέπει να σημειωθεί ότι, σε αυτό το διάγραμμα, η καμπύλη του Validation Loss παρουσιάζει μικρές διακυμάνσεις από την τέταρτη μέχρι την έβδομη εποχή. Αυτό σημαίνει το μοντέλο παρουσιάζει καλή εκμάθηση, αλλά ίσως δυσκολότερη γενίκευση για την μάσκα ZP, λόγω των μικρών διακυμάνσεων. Το μοντέλο φθάνει στην τιμή 0.4235 για το Training Loss και στην τιμή 0.4098 για το Validation Loss.



Εικόνα 5.7 – Διάγραμμα μεταβολής της απώλειας σε κάθε εποχή για την προβλεπόμενη μάσκα ICM (UNet)



Εικόνα 5.8 – Διάγραμμα μεταβολής της απώλειας σε κάθε εποχή για την προβλεπόμενη μάσκα TE (UNet)



Εικόνα 5.9 – Διάγραμμα μεταβολής της απώλειας σε κάθε εποχή για την προβλεπόμενη μάσκα ZP (UNet)

5.3 Αποτελέσματα για το μοντέλο DeepLab

Πραγματοποιήθηκε η διαδικασία εξαγωγής των προβλεπόμενων μασκών για διαφορετικούς συνδυασμούς των υπερπαραμέτρων εκπαίδευσης, με σκοπό την εύρεση των τιμών των υπερπαραμέτρων για τις οποίες το μοντέλο DeepLab παρουσιάζει την καλύτερη απόδοση. Πιο αναλυτικά, η τιμή των εποχών κυμαίνεται από 10 έως 20, η τιμή του learning rate από 0.0001 σε 0.001 και η τιμή του batch size από 8 σε 16, και από 16 σε 32. Στην συνέχεια, παρατίθεται ο Πίνακας 5.4, ο οποίος περιέχει τις τιμές της μετρικής αξιολόγησης IoU για κάθε πιθανό συνδυασμό των τιμών των υπερπαραμέτρων εκπαίδευσης.

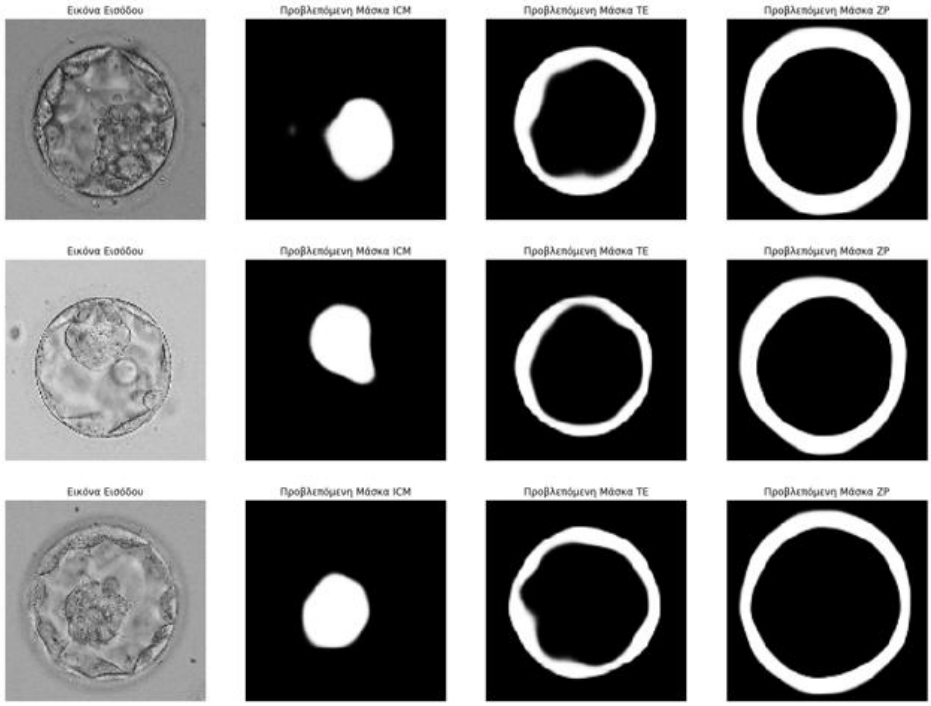
Πίνακας 5.4 – Αποτελέσματα των τιμών της μετρικής IoU για διαφορετικούς συνδυασμούς των υπερπαραμέτρων εκπαίδευσης του μοντέλου DeepLab

Παράμετροι Εκπαίδευσης	**Intersection of Union**		
	ICM	**TE**	**ZP**
Epochs = 10 Batch size = 8 Learning Rate = 0.0001	0.3616	0.2694	0.5091

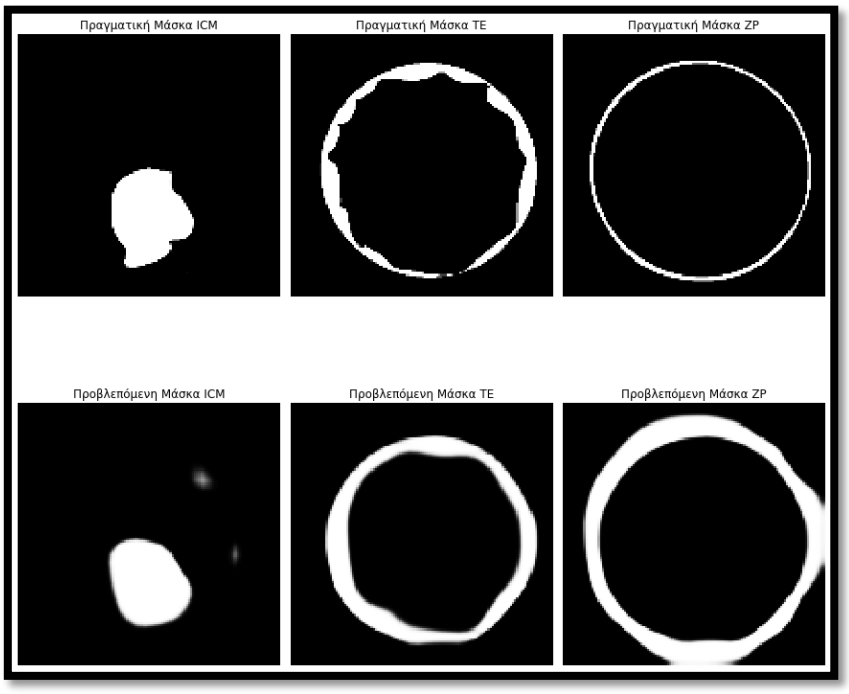
Epochs = 10 Batch size = 8 Learning Rate = 0.001	0.6148	0.4587	0.5889
Epochs = 10 Batch size = 16 Learning Rate = 0.0001	0.3982	0.2394	0.4616
Epochs = 10 Batch size = 16 Learning Rate = 0.001	0.3270	0.4240	0.6147
Epochs = 10 Batch size = 32 Learning Rate = 0.0001	0.5808	0.2637	0.4658
Epochs = 10 Batch size = 32 Learning Rate = 0.001	0.3421	0.2702	0.4900
Epochs = 20 Batch size = 8 Learning Rate = 0.0001	0.5437	0.4879	0.6338
Epochs = 20 Batch size = 8 Learning Rate = 0.001	0.6709	0.5149	0.6417
Epochs = 20 Batch size = 16 Learning Rate = 0.0001	0.3494	0.3747	0.5219
Epochs = 20 Batch size = 16 Learning Rate = 0.001	0.4729	0.4463	0.5904
Epochs = 20 Batch size = 32 Learning Rate = 0.0001	0.5379	0.2749	0.4925
Epochs = 20 Batch size = 32 Learning Rate = 0.001	0.5304	0.5329	0.6535

Μετά την παράθεση των αποτελεσμάτων του Πίνακα 5.4, γίνεται εύκολα αντιληπτό ότι το μοντέλο DeepLab παρουσιάζει την καλύτερη απόδοση για αριθμό εποχών ίσο με 20, τιμή του learning rate ίση με 0.001 και τιμή του batch size ίση με 8. Επομένως, ακολουθεί η παράθεση των εικόνων των προβλεπόμενων μασκών τμηματοποίησης για τα τρια χαρακτηριστικά ICM, TE και ZP, καθώς και των διαγραμμάτων της μετρικής αξιολόγησης IoU και της απώλειας κατά την διάρκεια των εποχών, για τον συγκεκριμένο συνδυασμό υπερπαραμέτρων. Στην Εικόνα 5.10, παρουσιάζεται ένα δείγμα από τις μάσκες που εξήχθησαν από το μοντέλο DeepLab. Πιο συγκεκριμένα, παρουσιάζεται ένα δείγμα από τις αρχικές εικόνες που δόθηκαν σαν είσοδος στο δίκτυο, καθώς και οι τρεις μάσκες ICM, TE και ZP, οι οποίες είναι οι έξοδοι του δικτύου για κάθε εικόνα εισόδου αντίστοιχα. Στις Εικόνες 5.11 και 5.12, παρουσιάζονται οι μάσκες ICM, TE, ZP που εξήχθησαν από το

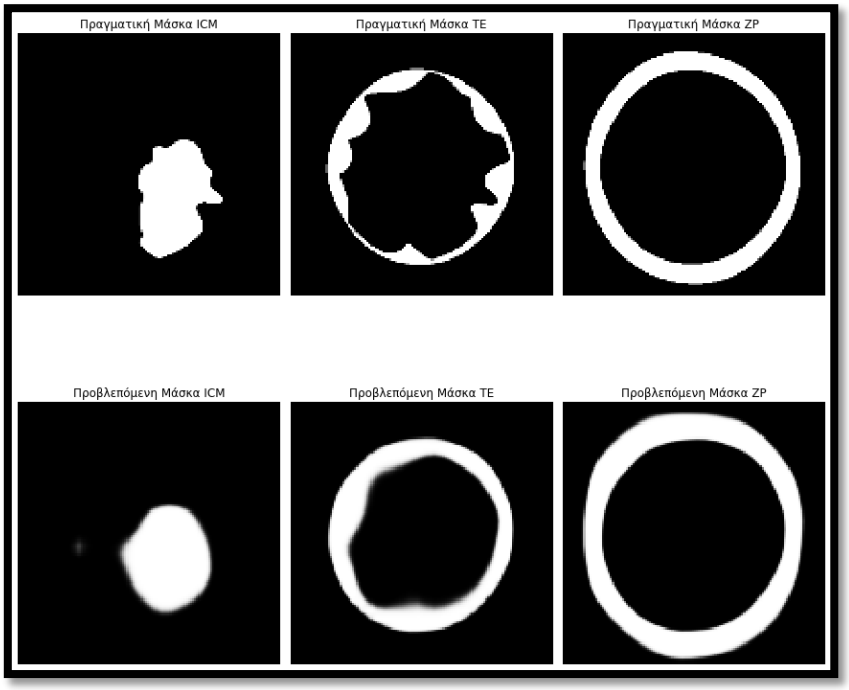
μοντέλο σε σύγκριση με τις πραγματικές μάσκες, για δύο τυχαίες εικόνες βλαστοκύστεων του συνόλου δεδομένων, με σκοπό την ανίχνευση των διαφορών τους.



Εικόνα 5.10 – Προβλεπόμενες μάσκες ICM, TE & ZP για το μοντέλο DeepLab



Εικόνα 5.11 – Σύγκριση προβλεπόμενων και πραγματικών μασκών ICM, TE & ZP για το μοντέλο DeepLab 1/2



Εικόνα 5.12 – Σύγκριση προβλεπόμενων και πραγματικών μασκών ICM, TE & ZP για το μοντέλο DeepLab 2/2

Επίσης, στον Πίνακα 5.5, παρουσιάζονται τα συγκεντρωτικά αποτελέσματα που εξήχθησαν για την μετρική αξιολόγησης IoU, καθ' όλη την διάρκεια της εκπαίδευσης και της επικύρωσης του μοντέλου, μετά από κάθε εποχή ενώ στον Πίνακα 5.6, παρουσιάζονται τα συγκεντρωτικά αποτελέσματα που εξήχθησαν για την απώλεια, καθόλη την διάρκεια της εκπαίδευσης και της επικύρωσης του μοντέλου, μετά από κάθε εποχή.

Πίνακας 5.5 – Συγκεντρωτικά αποτελέσματα της μετρικής αξιολόγησης IoU καθ' όλη την διάρκεια της εκπαίδευσης και της επικύρωσης του μοντέλου DeepLab, μετά από κάθε εποχή

Εποχές	**Training IoU**			**Validation IoU**		
	ICM	**TE**	**ZP**	**ICM**	**TE**	**ZP**
1	0.4915	0.2517	0.2854	0.4631	0.2683	0.4710
2	0.6879	0.4522	0.5061	0.2301	0.1821	0.4054
3	0.7284	0.4940	0.5821	0.5118	0.3779	0.4078
4	0.7577	0.5233	0.6320	0.4965	0.3253	0.4496
5	0.7722	0.5538	0.6461	0.5135	0.3237	0.3940
6	0.7863	0.5645	0.6669	0.5767	0.2309	0.4235

7	0.7872	0.5953	0.7113	0.5648	0.3866	0.5709
8	0.8071	0.5910	0.6942	0.6363	0.3685	0.5862
9	0.8107	0.6195	0.7326	0.6001	0.3636	0.4993
10	0.8200	0.6406	0.7420	0.6114	0.3789	0.5045
11	0.8228	0.6379	0.7549	0.5894	0.4002	0.4888
12	0.8314	0.6560	0.7570	0.6626	0.4181	0.5797
13	0.8378	0.6664	0.7621	0.6643	0.4677	0.6349
14	0.8268	0.6739	0.7976	0.6556	0.4685	0.6667
15	0.8399	0.6777	0.7891	0.6220	0.4938	0.6671
16	0.8459	0.6771	0.8033	0.6062	0.5036	0.6650
17	0.8370	0.6898	0.7806	0.6392	0.5197	0.6443
18	0.8490	0.6912	0.7832	0.6798	0.4750	0.6481
19	0.8536	0.6979	0.8079	0.6741	0.5360	0.6743
20	0.8561	0.6988	0.8034	0.6709	0.5149	0.6417

Πίνακας 5.6 – Συγκεντρωτικά αποτελέσματα της απώλειας καθ' όλη την διάρκεια της εκπαίδευσης και της επικύρωσης του μοντέλου DeepLab, μετά από κάθε εποχή

Εποχές	**Training Loss**			**Validation Loss**		
	ICM	**TE**	**ZP**	**ICM**	**TE**	**ZP**
1	0.5505	0.9050	0.9226	0.7798	1.0642	1.0151
2	0.2720	0.5861	0.5696	1.5235	1.4904	1.4309
3	0.2287	0.5187	0.4617	0.6898	1.0590	1.5334
4	0.2015	0.4822	0.3947	0.7365	1.2166	1.3056
5	0.1871	0.4396	0.3931	0.7410	1.1767	1.5682
6	0.1747	0.4277	0.3668	0.5845	1.5660	1.4632
7	0.1766	0.3847	0.3046	0.5838	0.9836	0.8638
8	0.1567	0.3998	0.3387	0.4778	0.9902	0.7370
9	0.1496	0.3605	0.2749	0.4780	1.0694	1.1101
10	0.1448	0.3400	0.2675	0.5159	1.0423	1.0740
11	0.1390	0.3454	0.2519	0.5774	0.9751	1.2111
12	0.1317	0.3214	0.2541	0.4295	0.8974	0.8001
13	0.1260	0.3101	0.2463	0.4089	0.8482	0.6246

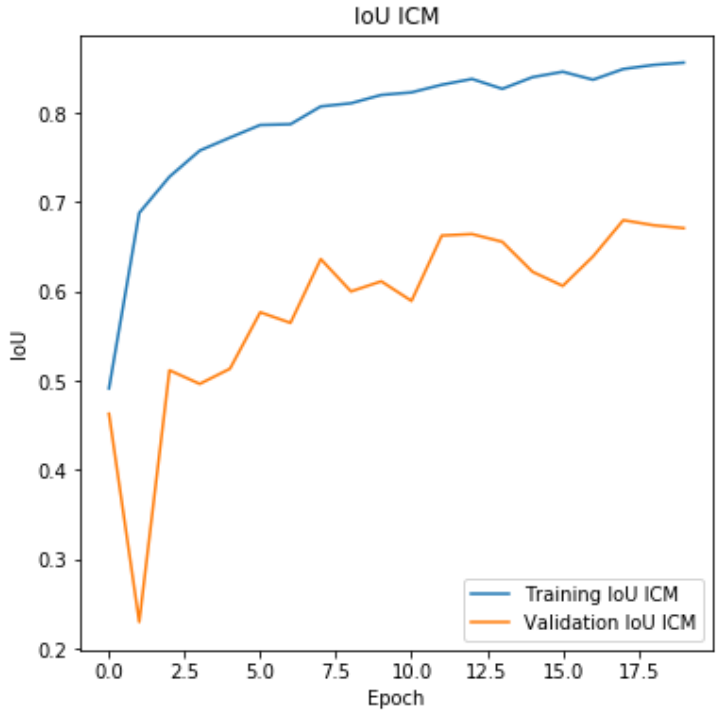
14	0.1394	0.3029	0.2027	0.4958	0.7475	0.5581
15	0.1262	0.3004	0.2167	0.4792	0.7517	0.5576
16	0.1179	0.3045	0.1949	0.4970	0.7711	0.6101
17	0.1304	0.2865	0.2320	0.4510	0.7184	0.6076
18	0.1184	0.2818	0.2281	0.3757	0.7705	0.5788
19	0.1128	0.2747	0.1930	0.3906	0.6189	0.5514
20	0.1108	0.2787	0.2039	0.3760	0.7289	0.6528

Κάνοντας χρήση των παραπάνω αποτελεσμάτων, εξήχθησαν τα διαγράμματα, τόσο της μετρικής αξιολόγησης IoU, όσο και της απώλειας, καθ' όλη την διάρκεια της εκπαίδευσης και της επικύρωσης του μοντέλου DeepLab, μετά από κάθε εποχή. Στις Εικόνες 5.13, 5.14 και 5.15, παρουσιάζονται τα διαγράμματα της μετρικής αξιολόγησης IoU, για κάθε προβλεπόμενη μάσκα ICM, TE, ZP.

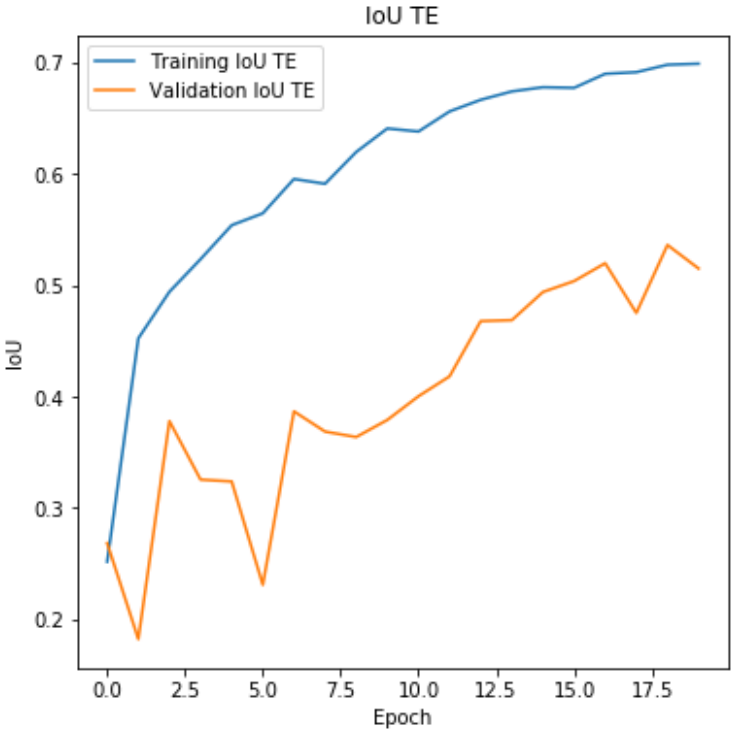
Η καμπύλη του Traning IoU για την μάσκα ICM ξεκινάει από την τιμή 0.4915, στην συνέχεια εμφανίζει μια γρήγορη αύξηση και φθάνει στην τιμή 0.8561. Από την άλλη, η καμπύλη του Validation IoU ξεκινάει με μια χαμηλότερη τιμή 0.4631, στην συνέχεια παρουσιάζει έντονες διακυμάνσεις και καταλήγει στην τιμή 0.6709. Πρέπει να σημειωθεί ότι παρουσιάζονται μεγάλες αποκλίσεις στις τιμές του Training IoU και του Validation IoU, το οποίο υποδηλώνει την πιθανή παρουσία overfitting.

Το διάγραμμα των καμπυλών του Training IoU και του Validation IoU για την μάσκα TE, παρουσιάζει μεγαλύτερες αποκλίσεις στις τιμές του Training IoU και του Validation IoU, το οποίο υποδηλώνει μεγαλύτερη εμφάνιση του φαινομένου overfitting. Η καμπύλη του Traning IoU για την μάσκα ICM ξεκινάει από την τιμή 0.2517, στην συνέχεια εμφανίζει μια γρήγορη αύξηση και φθάνει στην τιμή 0.6988. Από την άλλη, η καμπύλη του Validation IoU ξεκινάει με μια τιμή 0.2683, στην συνέχεια παρουσιάζει έντονες διακυμάνσεις και καταλήγει στην τιμή 0.5149.

Η καμπύλη του Traning IoU για την μάσκα ZP ξεκινάει από την τιμή 0.2854, αυξάνεται σταδιακά και φθάνει στην τιμή 0.8034. Από την άλλη, η καμπύλη του Validation IoU ξεκινάει με μια τιμή 0.4710, στην συνέχεια παρουσιάζει έντονες διακυμάνσεις και καταλήγει στην τιμή 0.6417. Πρέπει να σημειωθεί ότι συνεχίζουν να παρουσιάζονται μεγάλες αποκλίσεις στις τιμές του Training IoU και του Validation IoU, αλλά είναι μικρότερες συγκριτικά με αυτές που εμφανίζονται στο διάγραμμα των καμπυλών του Training IoU και του Validation IoU για την μάσκα TE.



Εικόνα 5.13 – Διάγραμμα μεταβολής του IoU σε κάθε εποχή για την προβλεπόμενη μάσκα ICM (DeepLab)

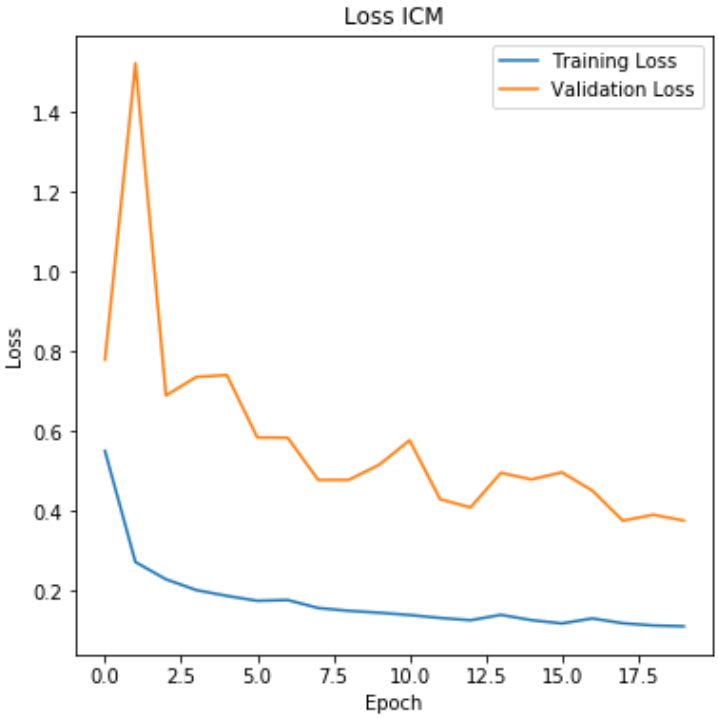


Εικόνα 5.14 – Διάγραμμα μεταβολής του IoU σε κάθε εποχή για την προβλεπόμενη μάσκα TE (DeepLab)

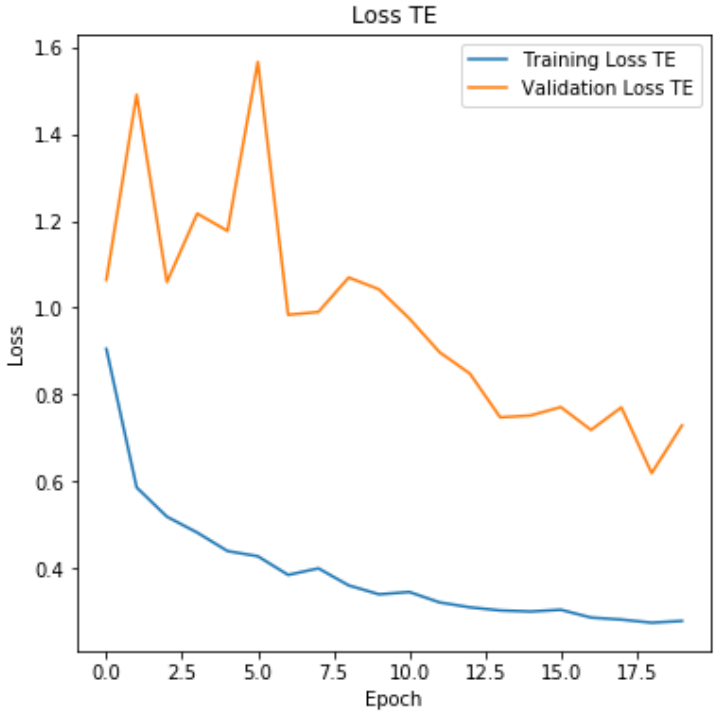
Εικόνα 5.15 – Διάγραμμα μεταβολής του IoU σε κάθε εποχή για την προβλεπόμενη μάσκα ZP (DeepLab)

Στις Εικόνες 5.16, 5.17 και 5.18, παρουσιάζονται τα διαγράμματα της απώλειας, για κάθε προβλεπόμενη μάσκα ICM, TE, ZP. Στο διάγραμμα της απώλειας για την μάσκα ICM, είναι εμφανές ότι η καμπύλη του Training Loss ξεκινάει από μια τιμή 0.5505, μειώνεται σταθερά με το πέρασμα των εποχών και καταλήγει στην τιμή 0.1108. Από την άλλη, η καμπύλη του Validation Loss ξεκινάει από μια τιμή 0.7798, εμφανίζει μια απότομη αύξηση στην δεύτερη εποχή, στην συνέχεια μειώνεται με αρκετές διακυμάνσεις και καταλήγει στην τιμή 0.3760. Πρέπει να σημειωθεί ότι υπάρχουν μεγάλες αποκλίσεις στις τιμές του Training Loss και του Validation Loss, καθώς οι τιμές του Validation Loss παραμένουν αρκετά υψηλές, το οποίο υποδηλώνει μια μικρή εμφάνιση του φαινομένου overfitting.

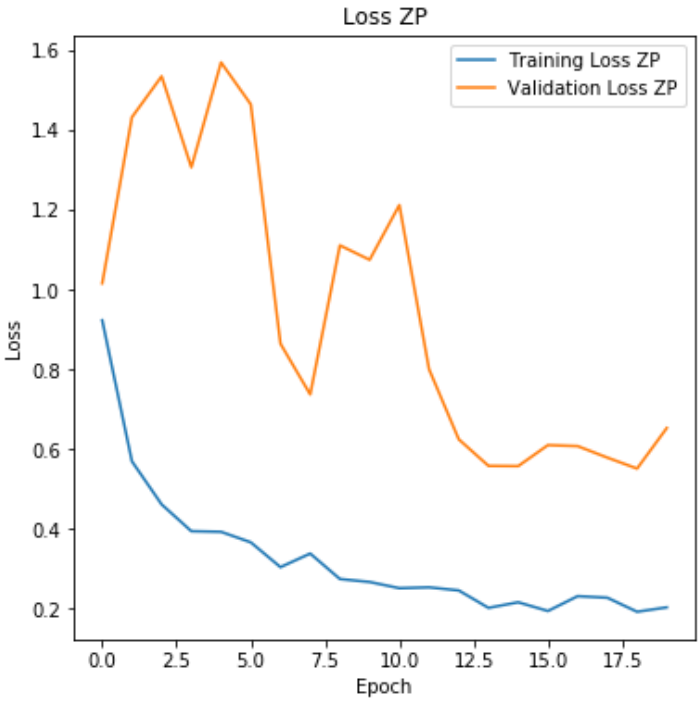
Όσον αφορά την απώλεια για την μάσκα TE, είναι εμφανές ότι η καμπύλη του Training Loss ξεκινάει από μια τιμή 0.9050, μειώνεται σταθερά με το πέρασμα των εποχών και καταλήγει στην τιμή 0.2787. Από την άλλη, η καμπύλη του Validation Loss ξεκινάει από μια τιμή 1.0642, στην συνέχεια εμφανίζει αρκετά έντονες διακυμάνσεις μέχρι την έβδομη εποχή, αμέσως μετά ξεκινάει να μειώνεται και καταλήγει στην τιμή 0.7289. Πρέπει να σημειωθεί ότι παρουσιάζεται αστάθεια σε σχέση με την κατηγορία ICM και εμφανίζεται μεγάλο πρόβλημα γενίκευσης για το μοντέλο.



Εικόνα 5.16 – Διάγραμμα μεταβολής της απώλειας σε κάθε εποχή για την προβλεπόμενη μάσκα ICM (DeepLab)



Εικόνα 5.17 – Διάγραμμα μεταβολής της απώλειας σε κάθε εποχή για την προβλεπόμενη μάσκα TE (DeepLab)



Εικόνα 5.18 – Διάγραμμα μεταβολής της απώλειας σε κάθε εποχή για την προβλεπόμενη μάσκα ZP (DeepLab)

Στο διάγραμμα της απώλειας για την μάσκα ZP, είναι εμφανές ότι η καμπύλη του Training Loss ξεκινάει από μια τιμή 0.9226, μειώνεται σταθερά με το πέρασμα των εποχών και καταλήγει στην τιμή 0.2039. Από την άλλη, η καμπύλη του Validation Loss ξεκινάει από μια τιμή 1.0151, στην συνέχεια εμφανίζει αρκετά έντονες διακυμάνσεις σε όλη την διάρκεια της μείωσή της και καταλήγει στην τιμή 0.6528. Πρέπει να σημειωθεί ότι οι διακυμάνσεις της καμπύλης Validation Loss για την μάσκα ZP είναι πιο έντονες από αυτές που παρουσιάστηκαν στις καμπύλες του Validation Loss για τις άλλες δύο κατηγορίες, ICM και TE, το οποίο υποδεικνύει αρκετά μεγαλύτερη αστάθεια και ανικανότητα για σύγκλιση.

5.4 Αποτελέσματα για το μοντέλο PSPNet

Πραγματοποιήθηκε η διαδικασία εξαγωγής των προβλεπόμενων μασκών για διαφορετικούς συνδυασμούς των υπερπαραμέτρων εκπαίδευσης, με σκοπό την εύρεση των τιμών των υπερπαραμέτρων για τις οποίες το μοντέλο PSPNet παρουσιάζει την

καλύτερη απόδοση. Πιο αναλυτικά, η τιμή των εποχών κυμαίνεται από 10 έως 20, η τιμή του learning rate από 0.0001 σε 0.001 και η τιμή του batch size από 8 σε 16, και από 16 σε 32. Στην συνέχεια, παρατίθεται ο Πίνακας 5.7, ο οποίος περιέχει τις τιμές της μετρικής αξιολόγησης IoU για κάθε πιθανό συνδυασμό των τιμών των υπερπαραμέτρων εκπαίδευσης.

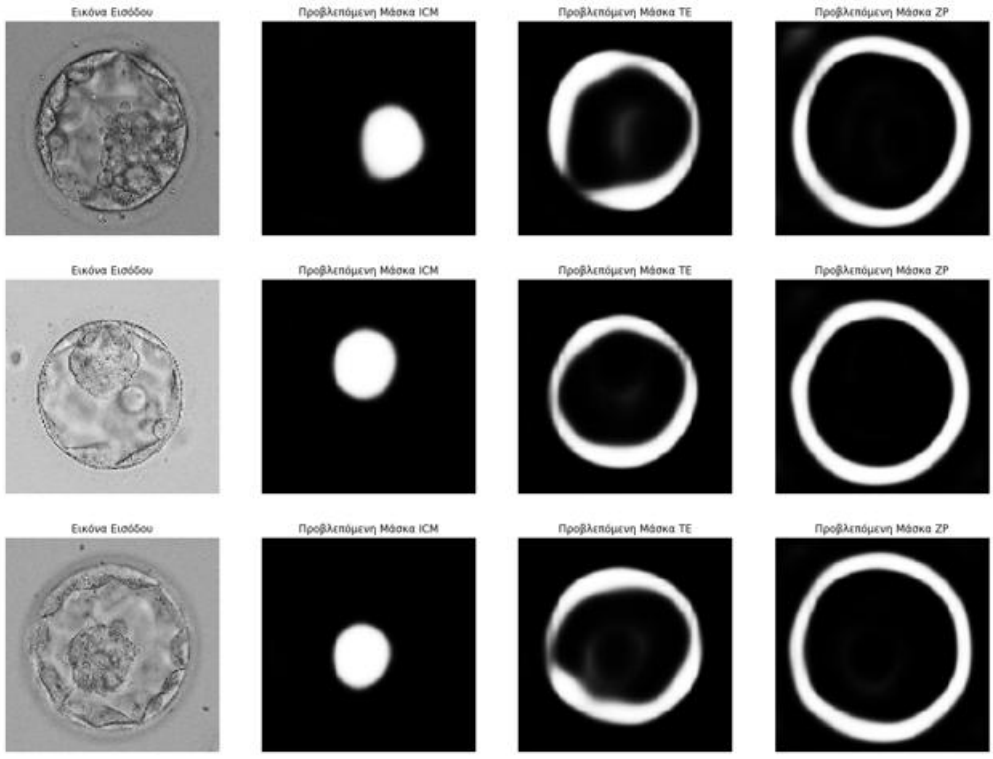
Πίνακας 5.7 – Αποτελέσματα των τιμών της μετρικής IoU για διαφορετικούς συνδυασμούς των υπερπαραμέτρων εκπαίδευσης του μοντέλου PSPNet

Παράμετροι Εκπαίδευσης	**Intersection of Union**		
	ICM	**TE**	**ZP**
Epochs = 10 Batch size = 8 Learning Rate = 0.0001	0.1827	0.1851	0.2343
Epochs = 10 Batch size = 8 Learning Rate = 0.001	0.5310	0.3566	0.4901
Epochs = 10 Batch size = 16 Learning Rate = 0.0001	0.1776	0.1732	0.2091
Epochs = 10 Batch size = 16 Learning Rate = 0.001	0.2715	0.3099	0.4135
Epochs = 10 Batch size = 32 Learning Rate = 0.0001	0.1433	0.1611	0.2291
Epochs = 10 Batch size = 32 Learning Rate = 0.001	0.1941	0.2213	0.3333
Epochs = 20 Batch size = 8 Learning Rate = 0.0001	0.3810	0.2752	0.3405
Epochs = 20 Batch size = 8 Learning Rate = 0.001	0.5770	0.4761	0.6143
Epochs = 20 Batch size = 16 Learning Rate = 0.0001	0.1941	0.1569	0.2130
Epochs = 20 Batch size = 16 Learning Rate = 0.001	0.4280	0.3991	0.5135
Epochs = 20 Batch size = 32 Learning Rate = 0.0001	0.1688	0.1747	0.2362
Epochs = 20 Batch size = 32 Learning Rate = 0.001	0.4282	0.2533	0.4458

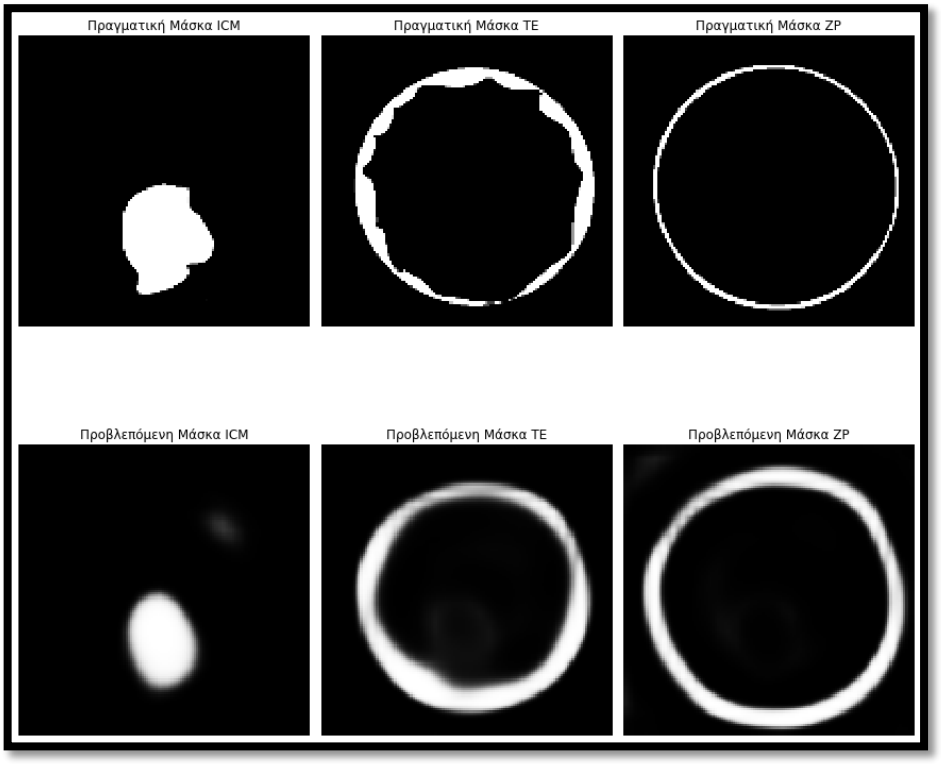
Μετά την παράθεση των αποτελεσμάτων του Πίνακα 5.7, γίνεται εύκολα αντιληπτό ότι το μοντέλο PSPNet παρουσιάζει την καλύτερη απόδοση για αριθμό εποχών ίσο με 20, τιμή του learning rate ίση με 0.001 και τιμή του batch size ίση με 8. Επομένως, ακολουθεί η παράθεση των εικόνων των προβλεπόμενων μασκών τμηματοποίησης για τα τρια χαρακτηριστικά ICM, TE και ZP, καθώς και των διαγραμμάτων της μετρικής αξιολόγησης IoU και της απώλειας κατά την διάρκεια των εποχών, για τον συγκεκριμένο συνδυασμό υπερπαραμέτρων.

Στην Εικόνα 5.19, παρουσιάζεται ένα δείγμα από τις μάσκες που εξήχθησαν από το μοντέλο PSPNet. Πιο συγκεκριμένα, παρουσιάζεται ένα δείγμα από τις αρχικές εικόνες που δόθηκαν σαν είσοδος στο δίκτυο, καθώς και οι τρεις μάσκες ICM, TE και ZP, οι οποίες είναι οι έξοδοι του δικτύου για κάθε εικόνα εισόδου αντίστοιχα.

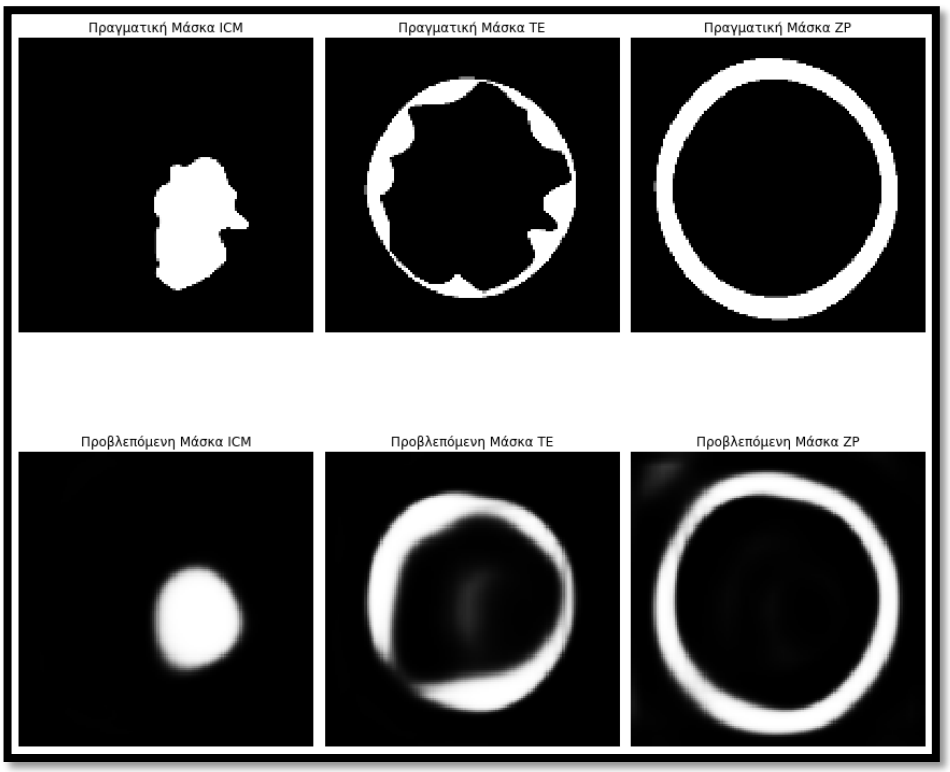
Στις Εικόνες 5.20 και 5.21, παρουσιάζονται οι μάσκες ICM, TE, ZP που εξήχθησαν από το μοντέλο σε σύγκριση με τις πραγματικές μάσκες, για δύο τυχαίες εικόνες βλαστοκύστεων του συνόλου δεδομένων, με σκοπό την ανίχνευση των διαφορών τους.



Εικόνα 5.19 – Προβλεπόμενες μάσκες ICM, TE & ZP για το μοντέλο PSPNet



Εικόνα 5.20 – Σύγκριση προβλεπόμενων και πραγματικών μασκών ICM, TE & ZP για το μοντέλο PSPNet 1/2



Εικόνα 5.21 – Σύγκριση προβλεπόμενων και πραγματικών μασκών ICM, TE & ZP για το μοντέλο PSPNet 2/2

Επίσης, στον Πίνακα 5.8, παρουσιάζονται τα συγκεντρωτικά αποτελέσματα που εξήχθησαν για την μετρική αξιολόγησης IoU, καθ' όλη την διάρκεια της εκπαίδευσης και της επικύρωσης του μοντέλου, μετά από κάθε εποχή ενώ στον Πίνακα 5.9, παρουσιάζονται τα συγκεντρωτικά αποτελέσματα που εξήχθησαν για την απώλεια, καθ' όλη την διάρκεια της εκπαίδευσης και της επικύρωσης του μοντέλου, μετά από κάθε εποχή.

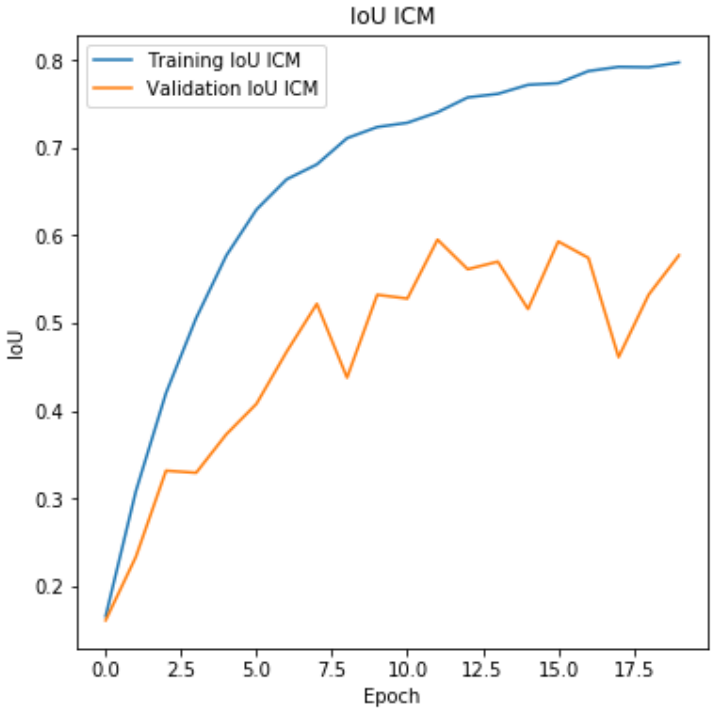
Πίνακας 5.8 – Συγκεντρωτικά αποτελέσματα της μετρικής αξιολόγησης IoU καθ' όλη την διάρκεια της εκπαίδευσης και της επικύρωσης του μοντέλου PSPNet, μετά από κάθε εποχή

Εποχές	**Training IoU**			**Validation IoU**		
	ICM	**TE**	**ZP**	**ICM**	**TE**	**ZP**
1	0.1655	0.1559	0.2050	0.1609	0.1884	0.2337
2	0.3079	0.2375	0.3114	0.2331	0.2680	0.3915
3	0.4200	0.2986	0.3939	0.3314	0.2961	0.4257
4	0.5056	0.3519	0.4554	0.3291	0.3193	0.4499
5	0.5768	0.3920	0.4985	0.3730	0.3828	0.5075
6	0.6293	0.4256	0.5417	0.4078	0.3875	0.4633
7	0.6638	0.4569	0.5737	0.4672	0.3729	0.5466
8	0.6807	0.4811	0.5944	0.5218	0.3915	0.4789
9	0.7107	0.5001	0.6166	0.4375	0.3790	0.5574
10	0.7234	0.5085	0.6150	0.5321	0.4641	0.5103
11	0.7282	0.5279	0.6452	0.5277	0.4196	0.4521
12	0.7402	0.5431	0.6632	0.5950	0.4779	0.5963
13	0.7572	0.5425	0.6523	0.5612	0.4056	0.4980
14	0.7612	0.5538	0.6631	0.5699	0.4646	0.4346
15	0.7715	0.5745	0.6977	0.5158	0.4274	0.4676
16	0.7734	0.5680	0.6952	0.5929	0.3966	0.3867
17	0.7872	0.5954	0.7110	0.5741	0.3521	0.5339
18	0.7920	0.6031	0.7113	0.4609	0.4174	0.4547
19	0.7916	0.6028	0.7059	0.5323	0.4990	0.5689
20	0.7970	0.6121	0.7326	0.5770	0.4761	0.6143

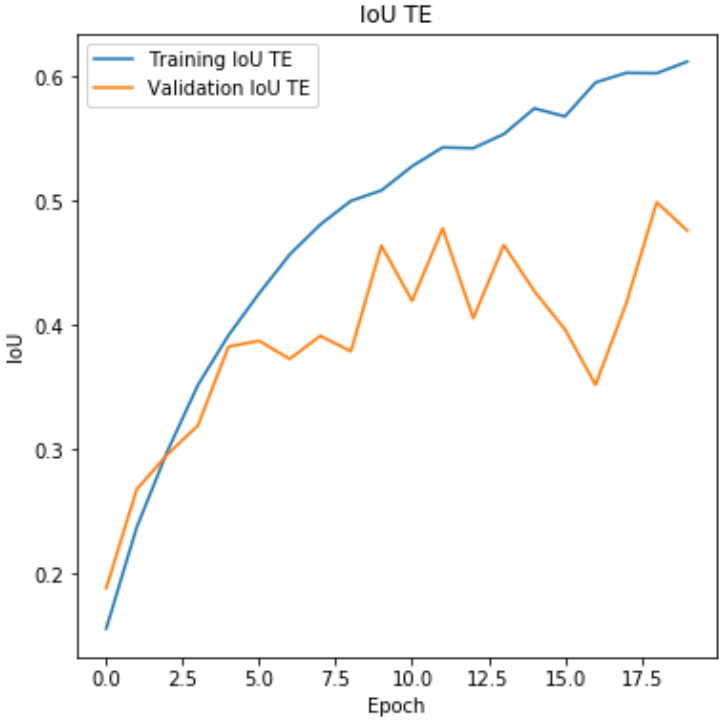
Πίνακας 5.9 – Συγκεντρωτικά αποτελέσματα της απώλειας καθ' όλη την διάρκεια της εκπαίδευσης και της επικύρωσης του μοντέλου PSPNet, μετά από κάθε εποχή

Εποχές	**Training Loss**			**Validation Loss**		
	ICM	**TE**	**ZP**	**ICM**	**TE**	**ZP**
1	1.2338	1.3176	1.2484	2.0774	1.4415	2.5299
2	0.7522	0.9541	0.8826	1.3725	0.8447	0.7921
3	0.5479	0.7985	0.7005	1.0158	0.8365	0.6784
4	0.4353	0.6964	0.5979	1.0694	0.7650	0.6566
5	0.3547	0.6314	0.5305	0.7643	0.6786	0.5770
6	0.2983	0.5763	0.4710	0.5888	0.6951	0.5978
7	0.2686	0.5399	0.4329	0.4911	0.6904	0.5282
8	0.2513	0.5064	0.4086	0.4273	0.6639	0.6167
9	0.2236	0.4851	0.3819	0.5090	0.7282	0.5039
10	0.2118	0.4788	0.3914	0.4141	0.6227	0.5469
11	0.2104	0.4519	0.3510	0.4631	0.6335	0.6515
12	0.2010	0.4337	0.3316	0.3720	0.5797	0.4526
13	0.1842	0.4374	0.3437	0.4192	0.7566	0.6233
14	0.1821	0.4248	0.3387	0.3911	0.6294	0.7003
15	0.1728	0.3977	0.2943	0.4859	0.6463	0.6407
16	0.1714	0.4093	0.2991	0.3735	0.7952	0.8030
17	0.1597	0.3734	0.2839	0.3758	0.9542	0.5410
18	0.1556	0.3669	0.2819	0.5295	0.7000	0.6622
19	0.1569	0.3687	0.2856	0.4412	0.5511	0.4721
20	0.1514	0.3568	0.2568	0.4030	0.5878	0.4356

Κάνοντας χρήση των παραπάνω αποτελεσμάτων, εξήχθησαν τα διαγράμματα, τόσο της μετρικής αξιολόγησης IoU, όσο και της απώλειας, καθ' όλη την διάρκεια της εκπαίδευσης και της επικύρωσης του μοντέλου PSPNet, μετά από κάθε εποχή. Στις Εικόνες 5.22, 5.23 και 5.24, παρουσιάζονται τα διαγράμματα της μετρικής αξιολόγησης IoU, για κάθε προβλεπόμενη μάσκα ICM, TE, ZP.



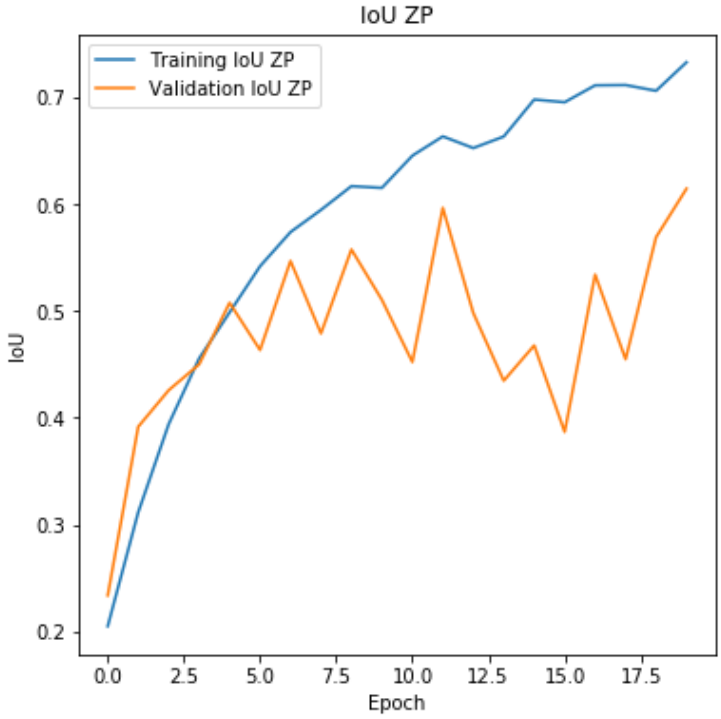
Εικόνα 5.22 – Διάγραμμα μεταβολής του IoU σε κάθε εποχή για την προβλεπόμενη μάσκα ICM (PSPNet)



Εικόνα 5.23 – Διάγραμμα μεταβολής του IoU σε κάθε εποχή για την προβλεπόμενη μάσκα TE (PSPNet)

Στο διάγραμμα των καμπυλών του Training IoU και του Validation IoU για την μάσκα ICM, είναι εμφανές ότι η καμπύλη του Training IoU ξεκινάει από την τιμή 0.1655, αυξάνεται σταδιακά και καταλήγει στην τιμή 0.7970. Από την άλλη, η καμπύλη του Validation IoU ξεκινάει από την τιμή 0.1609, εμφανίζει έντονες διακυμάνσεις μεταξύ των τιμών 0.4 και 0.6 και καταλήγει στην τιμή 0.5770. Μετά την δεύτερη με τρίτη εποχή, οι διαφορές των τιμών IoU μεταξύ των δύο καμπυλών, Training IoU και Validation IoU, αρχίζουν και αυξάνουν και σε συνδυασμό με τις έντονες διακυμάνσεις της καμπύλης του Validation IoU, εξάγεται το συμπέρασμα ότι το μοντέλο οδηγείται σε πιθανή εμφάνιση overfitting.

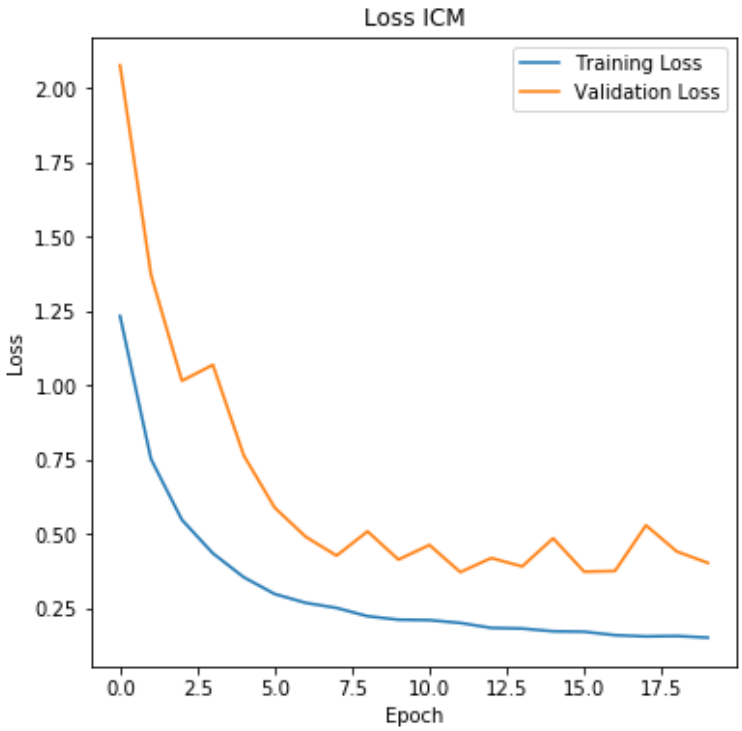
Στο διάγραμμα των καμπυλών του Training IoU και του Validation IoU για την μάσκα TE, είναι εμφανές ότι η καμπύλη του Training IoU ξεκινάει από την τιμή 0.1559, αυξάνεται σταδιακά και καταλήγει στην τιμή 0.6121. Από την άλλη, η καμπύλη του Validation IoU βρίσκεται πιο χαμηλά. Πιο συγκεκριμένα, ξεκινάει από την τιμή 0.1884, εμφανίζει έντονες διακυμάνσεις μεταξύ των τιμών 0.3 και 0.5 και καταλήγει στην τιμή 0.4761. Μπορεί να γίνει εύκολα αντιληπτό ότι η καμπύλη του Validation IoU παρουσιάζει πιο έντονες διακυμάνσεις, σε σχέση με την καμπύλη του Validation IoU στο αντίστοιχο διάγραμμα της μάσκας ICM. Σε αυτήν την περίπτωση, εξάγεται το συμπέρασμα ότι το μοντέλο παρουσιάζει αστάθεια και δυσκολεύεται να γενικεύσει σε ένα άλλο σύνολο δεδομένων.



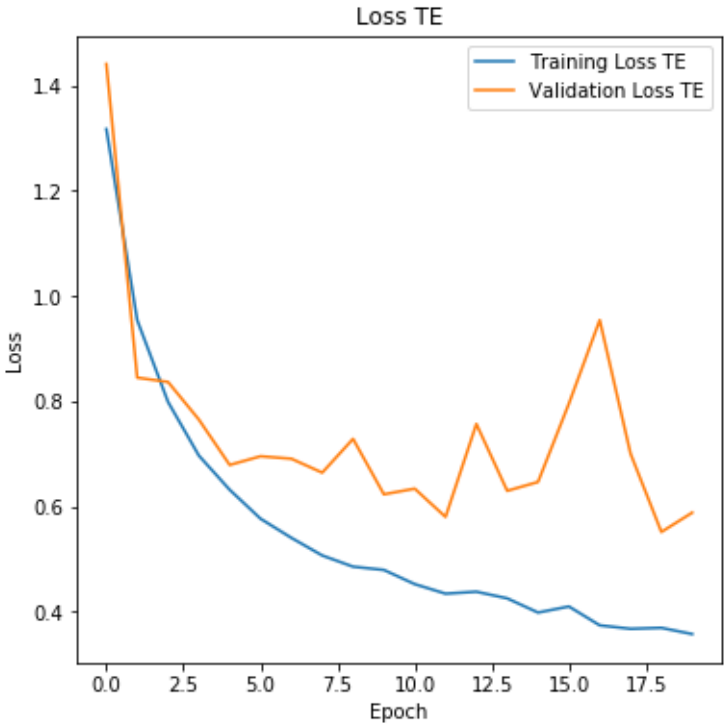
Εικόνα 5.24 – Διάγραμμα μεταβολής του IoU σε κάθε εποχή για την προβλεπόμενη μάσκα ZP (PSPNet)

Στο διάγραμμα των καμπυλών του Training IoU και του Validation IoU για την μάσκα ZP, είναι εμφανές ότι η καμπύλη του Training IoU ξεκινάει από την τιμή 0.2050, αυξάνεται σταδιακά και καταλήγει στην τιμή 0.7326. Από την άλλη, η καμπύλη του Validation IoU ξεκινάει από την τιμή 0.2337, εμφανίζει έντονες διακυμάνσεις και καταλήγει στην τιμή 0.6143. Πρέπει να σημειωθεί ότι οι διακυμάνσεις ξεκινούν από την πέμπτη εποχή και διατηρούνται μέχρι την δέκατη όγδοη εποχή. Το μοντέλο PSPNet παρουσιάζει καλύτερη συμπεριφορά για την κατηγορία ZP, σε σχέση με την κατηγορία TE, αλλά χρειάζεται βελτίωση όσον αφορά την γενίκευσή του.

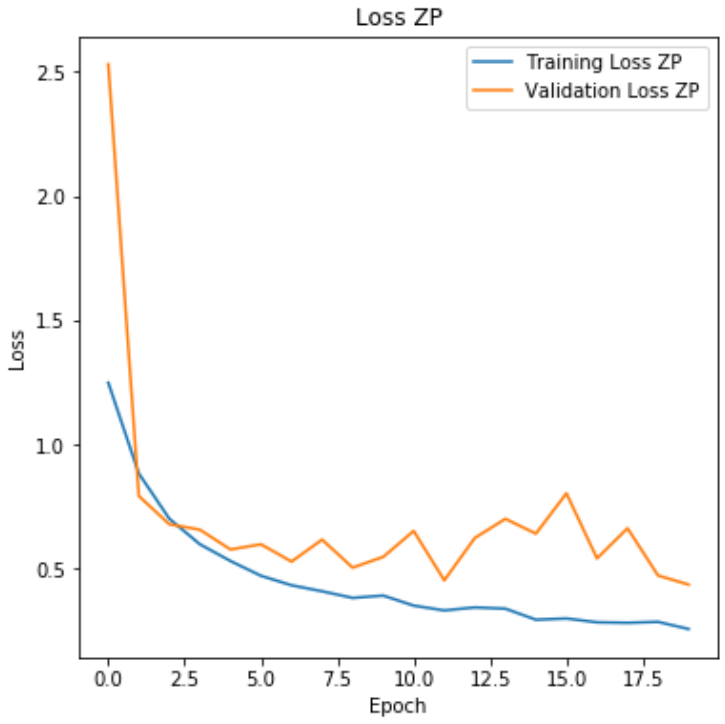
Στις Εικόνες 5.25, 5.26 και 5.27, παρουσιάζονται τα διαγράμματα της απώλειας, για κάθε προβλεπόμενη μάσκα ICM, TE, ZP. Στο διάγραμμα της απώλειας για την μάσκα ICM, είναι εμφανές ότι η καμπύλη του Training Loss ξεκινάει από μια τιμή 1.2338, μειώνεται σταθερά με το πέρασμα των εποχών και καταλήγει στην τιμή 0.1514. Από την άλλη, η καμπύλη του Validation Loss ξεκινάει από μια τιμή 2.0774, στην συνέχεια μειώνεται με ορισμένες μικρές διακυμάνσεις και καταλήγει στην τιμή 0.4030. Πρέπει να σημειωθεί ότι υπάρχουν μικρές αποκλίσεις στις τιμές του Training Loss και του Validation Loss, αλλά δεν είναι τόσο έντονο το φαινόμενο του overfitting.



Εικόνα 5.25 – Διάγραμμα μεταβολής της απώλειας σε κάθε εποχή για την προβλεπόμενη μάσκα ICM (PSPNet)



Εικόνα 5.26 – Διάγραμμα μεταβολής της απώλειας σε κάθε εποχή για την προβλεπόμενη μάσκα TE (PSPNet)



Εικόνα 5.27 – Διάγραμμα μεταβολής της απώλειας σε κάθε εποχή για την προβλεπόμενη μάσκα ZP (PSPNet)

Όσον αφορά την απώλεια για την μάσκα TE, είναι εμφανές ότι η καμπύλη του Training Loss ξεκινάει από μια τιμή 1.3176, μειώνεται σταθερά με το πέρασμα των εποχών και καταλήγει στην τιμή 0.3568. Από την άλλη, η καμπύλη του Validation Loss ξεκινάει από μια τιμή 1.4415, στην συνέχεια μειώνεται και μετά την δέκατη εποχή εμφανίζει έντονες διακυμάνσεις καταλήγοντας στην τιμή 0.5878. Πρέπει να σημειωθεί ότι οι έντονες διακυμάνσεις του Validation Loss υποδηλώνουν πιθανή εμφάνιση overfitting, δηλαδή πρόβλημα γενίκευσης από το μοντέλο.

Στο διάγραμμα της απώλειας για την μάσκα ZP, είναι εμφανές ότι η καμπύλη του Training Loss ξεκινάει από μια τιμή 1.2484, μειώνεται σταθερά και καταλήγει στην τιμή 0.2568. Από την άλλη, η καμπύλη του Validation Loss ξεκινάει από μια τιμή 2.5299, στην συνέχεια μειώνεται, αλλά εμφανίζει ορισμένες διακυμάνσεις και καταλήγει στην τιμή 0.4356. Πρέπει να σημειωθεί ότι οι αποκλίσεις μεταξύ των καμπυλών Training Loss και Validation Loss είναι πιο μικρές, σε σχέση με αυτές που εμφανίζονται στο αντίστοιχο διάγραμμα της μάσκας TE. Παρόλα αυτά, το μοντέλο τείνει προς την εμφάνιση overfitting και σε αυτήν την περίπτωση.

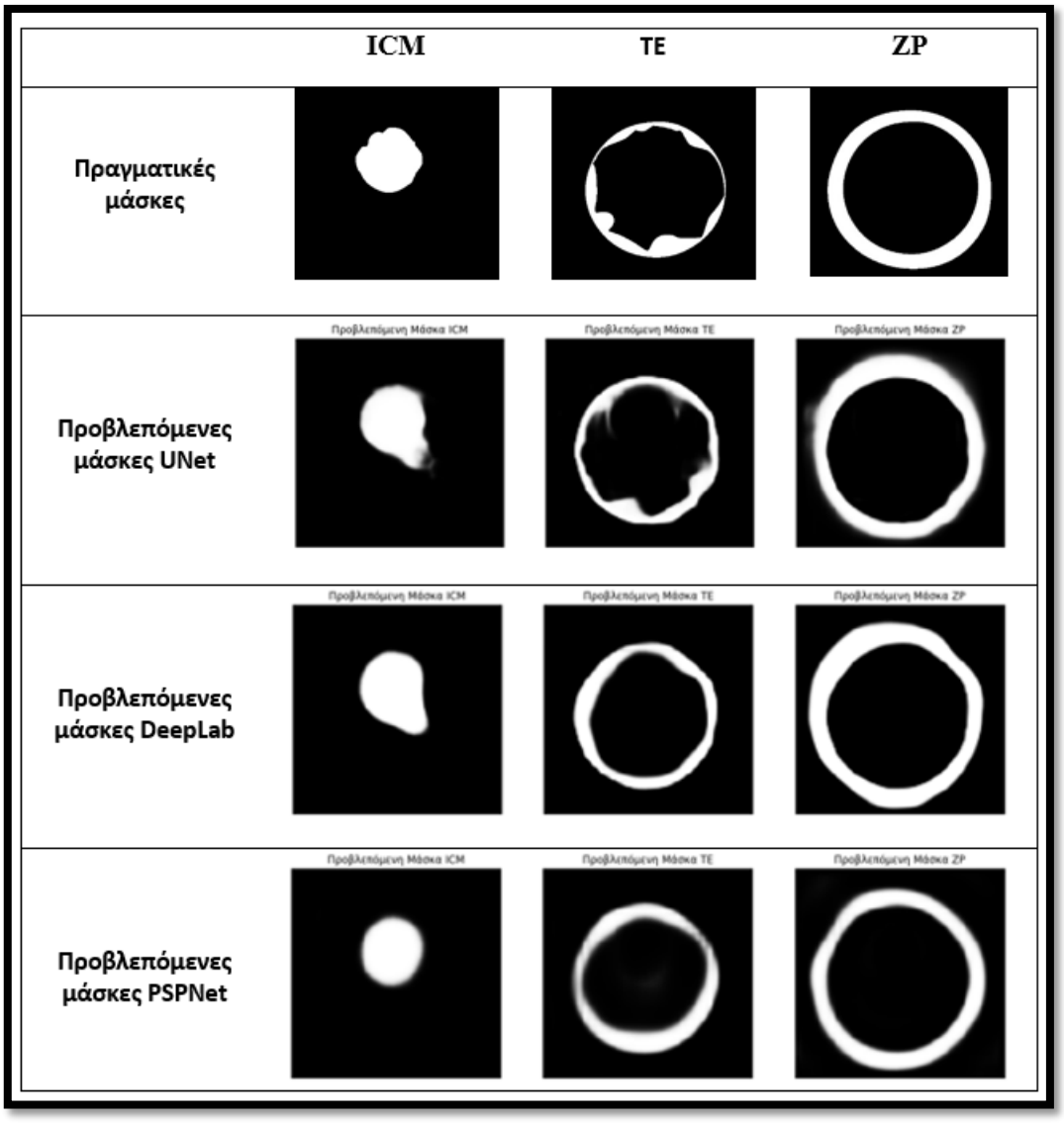
5.5 Σύγκριση Αποτελεσμάτων

Στον Πίνακα 5.10, παρουσιάζονται τα συγκεντρωτικά αποτελέσματα των τιμών της μετρικής αξιολόγησης IoU για τα μοντέλα UNet, DeepLab και PSPNet. Με βάσει τα παραπάνω αποτελέσματα και τα αποτελέσματα του Πίνακα 5.10, καταλήγουμε στο συμπέρασμα ότι το μοντέλο UNet ανταποκρίνεται καλύτερα στην πρόβλεψη των τριών μασκών ICM, TE και ZP.

Πίνακας 5.10 – Συγκεντρωτικά αποτελέσματα των τιμών της μετρικής αξιολόγησης IoU για τα μοντέλα UNet, DeepLab & PSPNet

Μοντέλα	**Intersection of Union**		
	ICM	**TE**	**ZP**
UNet	0.6705	0.5688	0.6365
DeepLab	0.6709	0.5149	0.6417
PSPNet	0.5770	0.4761	0.6143

Επιπλέον, στην Εικόνα 5.28 παρουσιάζεται η σύγκριση των προβλεπόμενων μασκών που εξήχθησαν από κάθε μοντέλο, για τα τρία χαρακτηριστικά ICM, TE και ZP. Παρατηρείται ότι το μοντέλο UNet παρουσιάζει την καλύτερη επίδοση, αν και διαθέτει ορισμένες ατέλειες. Από την άλλη, το μοντέλο DeepLab έχει πιο καθαρά όρια, αλλά εμφανίζει ελλείψεις, ιδιαίτερα για την μάσκα TE. Τέλος, το μοντέλο PSPNet έχει την χειρότερη επίδοση σε σχέση με τα άλλα δύο μοντέλα, εξαιτίας της εμφάνισης αρκετών ελλείψεων.



Εικόνα 5.28 – Σύγκριση πραγματικών και προβλεπόμενων μασκών από τα τρία μοντέλα UNet, DeepLab και PSPNet

6. Συζήτηση – Συμπεράσματα

Η προσπάθεια επίτευξης του στόχου της παρούσας Διπλωματικής εργασίας, δηλαδή της εξαγωγής των προβλεπόμενων μασκών ICM, TE και ZP, μέσω των τριών μοντέλων σημασιολογικής τμηματοποίησης UNet, DeepLab και PSPNet, οδήγησε σε ορισμένα σημαντικά συμπεράσματα.

Αρχικά, το μοντέλο UNet παρουσιάζει συνολικά την καλύτερη απόδοση, σε σύγκριση με τα άλλα δύο μοντέλα. Οι καμπύλες, τόσο της μετρικής αξιολόγησης IoU, όσο και της απώλειας κατά την διάρκεια της εκπαίδευσης και της επικύρωσης του μοντέλου UNet, ακολουθούν η μια την άλλη, χωρίς μεγάλες αποκλίσεις ή τυχόν έντονες διακυμάνσεις, το οποίο αντικατοπτρίζει την συμπεριφορά ενός σταθερού μοντέλου, με καλή γενίκευση και χωρίς προβλήματα υπερποσαρμογής.

Από την άλλη πλευρά, το μοντέλο DeepLab δεν παρουσιάζει την ίδια συμπεριφορά, αν και οι τιμές της μετρικής αξιολόγησης IoU είναι πολύ κοντά σε σχέση με τις αντίστοιχες τιμές του μοντέλου UNet. Οι καμπύλες, τόσο της μετρικής αξιολόγησης IoU, όσο και της απώλειας κατά την διάρκεια της εκπαίδευσης και της επικύρωσης του μοντέλου DeepLab, δεν ακολουθούν η μια την άλλη. Σε αυτήν την περίπτωση, παρουσιάζονται μεγάλες αποκλίσεις μεταξύ των καμπυλών και εμφανίζονται έντονες διακυμάνσεις, το οποίο αντικατοπτρίζει την συμπεριφορά ενός λιγότερο σταθερού μοντέλου που υποφέρει από το φαινόμενο της υπερπροσαρμογής.

Το μοντέλο PSPNet εμφανίζει την χαμηλότερη επίδοση, δηλαδή τις χαμηλότερες τιμές της μετρικής αξιολόγησης IoU, σε σύγκριση με τα άλλα δύο μοντέλα, UNet και DeepLab. Οι καμπύλες, τόσο της μετρικής αξιολόγησης IoU, όσο και της απώλειας κατά την διάρκεια της εκπαίδευσης και της επικύρωσης του μοντέλου PSPNet, δεν έχουν την ίδια πορεία. Παρουσιάζονται μεγάλες αποκλίσεις στις τιμές τους, καθώς και έντονες διακυμάνσεις, το οποίο αντικατοπτρίζει την συμπεριφορά ενός ασταθούς μοντέλου με δυσκολία στην γενίκευση, υποφέροντας από το φαινόμενο της υπερπροσαρμογής.

Τέλος, αναφέρονται ορισμένες προτάσεις και βελτιώσεις, οι οποίες πιθανόν θα οδηγήσουν στην επίτευξη καλύτερων αποτελεσμάτων στο τομέα της σημασιολογικής τμηματοποίησης ιατρικών εικόνων, και πιο συγκεκριμένα, εικόνων βλαστοκύστεων. Κρίνεται απαραίτητη η επίλυση ενός ιδιαίτερα σημαντικού προβλήματος, όπως η έλλειψη επαρκών συνόλων δεδομένων για την επίτευξη τέτοιου είδους επιστημονικών ερευνών. Η δυσκολία εύρεσης επαρκών συνόλων δεδομένων οδηγεί τις εν λόγω έρευνες σε περιορισμούς, καθώς υπάρχει αδυναμία ελέγχου ενός υλοποιημένου μοντέλου σε νέα δεδομένα. Ένα μοντέλο δεν μπορεί να θεωρηθεί αξιόπιστο, αν δεν έχει πραγματοποιηθεί

έλεγχος της συμπεριφοράς του σε νέα δεδομένα. Είναι αναγκαίο διάφοροι επιστήμονες του ιατρικού κλάδου να συνεισφέρουν στην αντιμετώπιση του εν λόγω προβλήματος, παρέχοντας επαρκή σύνολα δεδομένων.

Όσον αφορά τα μοντέλα DeepLab και PSPNet, τα οποία εμφανίστηκαν ως πιο ασταθή μοντέλα, λόγω των έντονων διακυμάνσεων στις καμπύλες της μετρικής αξιολόγησης IoU και της απώλειας, χωρίς καλή γενίκευση, επηρεασμένα από το φαινόμενο της υπερποσαρμογής, προτείνονται ορισμένες βελτιώσεις, οι οποίες πιθανόν θα οδηγήσουν σε καλύτερες αποδόσεις και εξάλειψη της υπερποσαρμογής.

Αρχικά, προτείνεται η αύξηση των εποχών, η οποία μπορεί να οδηγήσει σε καλύτερη σύγκλιση, ή μείωση του ρυθμού εκμάθησης, με σκοπό την εξασθένιση των διακυμάνσεων, άρα και την σταθερότερη σύγκλιση. Επιπλέον, θα μπορούσαν να δοκιμαστούν διάφορες τεχνικές για επίτευξη καλύτερης γενίκευσης των δύο μοντέλων. Ορισμένες από αυτές τις τεχνικές είναι το data augmentation, δηλαδή η αύξηση των δεδομένων μετά από διάφορους μετασχηματισμούς, ή το early stopping, το οποίο θα σταματούσε την εκπαίδευση των μοντέλων όταν τα αποτελέσματα του συνόλου επικύρωσης θα ξεκινούσαν να μην είναι και τόσο καλά. Οι δύο παραπάνω τεχνικές μπορεί να οδηγήσουν σε αποφυγή της υπερπροσαρμογής άρα καλύτερη γενίκευση για τα μοντέλα DeepLab και PSPNet. Το Regularization και το Batch Normalization είναι επίσης δύο καλές τεχνικές που θα μπορούσαν να χρησιμοποιηθούν για την επίτευξη καλύτερων αποτελεσμάτων. Πρέπει να διευκρινιστεί ότι όλα τα παραπάνω προυποθέτουν την ύπαρξη αρκετών υπολογιστικών πόρων, το οποίο είναι άξιο σημασίας και θα πρέπει να πραγματοποιηθούν ορισμένες δοκιμές σε διαφορετικά υπολογιστικά περιβάλλοντα.

Αναφορές

Scott, R. T., Jr., et al. (2012). Comprehensive chromosome screening is highly predictive of the reproductive potential of human embryos: A prospective, blinded, nonselection study. *Fertility and Sterility, 97*, 870–875.

Werner, M. D., et al. (2014). Clinically recognizable error rate after the transfer of comprehensive chromosomal screened euploid embryos is low. *Fertility and Sterility, 102*(6), 1613–1618.

Shapiro, L. G., & Stockman, G. C. (2001). *Computer vision*. Prentice-Hall. ISBN 0-13-030796-3.

Barghout, L., & Lee, L. W. (2003). Perceptual information processing system. *U.S. Patent Application No. 10/618,543*. Paravue Inc. Filed July 11, 2003.

Guo, D., Pei, Y., Zheng, K., Yu, H., Lu, Y., & Wang, S. (2020). Degraded image semantic segmentation with dense-gram networks. *IEEE Transactions on Image Processing, 29*, 782–795.

Saeedi, P. (2017). Automatic identification of human blastocyst components via texture. *IEEE Transactions on Biomedical Engineering, 64*(12), 2908–2918.

Zhang, Z., Liu, Q., & Wang, Y. (2017). Road extraction by deep residual U-Net.

Ibtehaz, N., & Rahman, M. S. (2019). MultiResUNet: Rethinking the U-Net architecture for multimodal biomedical image segmentation.

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation. *Proceedings of the Medical Image Computing and Computer-Assisted Intervention Conference (MICCAI 2015)*.

Long, J., Shelhamer, E., & Darrell, T. (2017). Fully convolutional networks for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 39*(4), 640–651.

Milletari, F., Navab, N., & Ahmadi, S. A. (2016). V-Net: Fully convolutional neural networks for volumetric medical image segmentation. *Conference on 3D Vision (3DV).*

Oktay, O., Schlemper, J., Le Folgoc, L., Lee, M., Heinrich, M., Misawa, K., Mori, K., McDonagh, S., Hammerla, N. Y., Kainz, B., Glocker, B., & Rueckert, D. (2018). Attention U-Net: Learning where to look for the pancreas.

Alom, M. Z., Hasan, M., Yakopcic, C., Taha, T. M., & Asari, V. K. (2018). Recurrent residual convolutional neural network based on U-Net (R2U-Net) for medical image segmentation.

Jha, D., Smedsrud, P. H., Riegler, M. A., Johansen, D., de Lange, T., Halvorsen, P., & Johansen, H. D. (2019). ResUNet++: An advanced architecture for medical image segmentation.

Zhao, H., Shi, J., Qi, X., Wang, X., & Jia, J. (2017). Pyramid scene parsing network. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 39*(4), 851–864.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2014). Semantic image segmentation with deep convolutional nets and fully connected CRFs.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2016). DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs.

Chen, L.-C., Papandreou, G., Schroff, F., & Adam, H. (2017). Rethinking atrous convolution for semantic image segmentation.

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., & Adam, H. (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation. *In Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 801–818).

Gurney, K. (2018). An introduction to neural networks. CRC press.

Grossi, E., & Buscema, M. (2007). Introduction to artificial neural networks. *European Journal of Gastroenterology & Hepatology, 19*(12), 1046–1054.

Rojas, R. (2013). *Neural networks: A systematic introduction*. Springer Science & Business Media.

Yacim, J. A., & Boshoff, D. (2018). Impact of artificial neural networks training algorithms on accurate prediction of property values. *Journal of Real Estate Research, 40*(3), 375–418.

M'ng, J. C. P., & Mehralizadeh, M. (2016). Forecasting East Asian indices futures via a novel hybrid of wavelet-PCA denoising and artificial neural network models. *PLOS ONE, 11*(6), e0156338.

O'Shea, K., & Nash, R. (2015). An introduction to convolutional neural networks.

Lo, C. C., Lee, C. H., & Huang, W. C. (2020). Prognosis of bearing and gear wears using convolutional neural network with hybrid loss function. *Sensors, 20*(12), 3539.

Mylonas, P., Spyrou, E., Avrithis, Y., & Kollias, S. (2009). Using visual context and region semantics for high-level concept detection. *IEEE Transactions on Multimedia, 11*(2), 229–243.

Long, J., Shelhamer, E., & Darrell, T. (2014). Fully convolutional networks for semantic segmentation.

Tealab, A. (2018). Time series forecasting using artificial neural networks methodologies: A systematic review. *Future Computing and Informatics Journal, 3*(2), 334–340.

Sak, H., Senior, A., & Beaufays, F. (2014). Long short-term memory recurrent neural network architectures for large scale acoustic modeling.

Li, X., & Wu, X. (2014). Constructing long short-term memory based deep recurrent neural networks for large vocabulary speech recognition.

Graves, A., Liwicki, M., Fernandez, S., Bertolami, R., Bunke, H., & Schmidhuber, J. (2009). A novel connectionist system for improved unconstrained handwriting recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 31*(5), 855–868.

Kumar, N., Agarwal, P., Bansal, S., Yadav, V. K., & Bhowmik, D. (2024). Sentiment analysis using novel deep learning methods. *Proceedings on Engineering Sciences, 6*(2), 853–862.

Nosouhian, F. K., & Nosouhian, S. (2021). A review of recurrent neural network architecture for sequence learning: Comparison between LSTM and GRU.