



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

UNIVERSITY OF PIRAEUS



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
ΤΜΗΜΑ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

Π.Μ.Σ.
ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ

ΣΤΑΥΡΙΑΝΟΣ ΚΩΝΣΤΑΝΤΙΝΟΣ

Διπλωματική Εργασία

ΑΝΑΓΝΩΡΙΣΗ ΔΙΑΤΑΞΗΣ ΣΧΗΜΑΤΙΣΜΟΥ
ΑΘΛΗΤΙΚΩΝ ΟΠΤΙΚΩΝ ΔΕΔΟΜΕΝΩΝ

Πειραιάς 2022

Υποβάλλεται
για την εκπλήρωση των προϋποθέσεων λήψης
Μεταπτυχιακού Διπλώματος
στην «Τεχνητή Νοημοσύνη»
στο
ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

Submitted
in partial fulfilment of the requirements for the degree of
Master of Artificial Intelligence
at the
UNIVERSITY OF PIRAEUS

February 2022

Πανεπιστήμιο Πειραιώς, ΕΚΕΦΕ «ΔΗΜΟΚΡΙΤΟΣ». Κάτοχος όλων των δικαιωμάτων

/ University of Piraeus, NCSR "Demokritos". All rights reserved.

Συγγραφέας: Σταυριανός Κωνσταντίνος

Έγκριση από την εξεταστική επιτροπή

(Υπογραφή)

(Υπογραφή)

(Υπογραφή)

.....

.....

.....

Ηλίας Μαγκλογιάννης
Καθηγητής

Θεόδωρος Γιαννακόπουλος
Ερευνητής Β΄

Μιχαήλ Φιλιππάκης
Καθηγητής

**ΑΝΑΓΝΩΡΙΣΗ ΔΙΑΤΑΞΗΣ ΣΧΗΜΑΤΙΣΜΟΥ
ΑΘΛΗΤΙΚΩΝ ΟΠΤΙΚΩΝ ΔΕΔΟΜΕΝΩΝ**

Σταυριανός Κωνσταντίνος

Υποβλήθηκε στο ΔΠΜΣ «Τεχνητή Νοημοσύνη» την XX Μηνός 20XX ως
υποχρέωση για την λήψη Μεταπτυχιακού Διπλώματος Σπουδών

Submitted to the II-MSc “Artificial Intelligence” on Month XX, 20XX, in
partial fulfillment of the
requirements for the MSc degree

ΕΥΧΑΡΙΣΤΙΕΣ

Τα λόγια είναι φτωχά και δεν μπορούν να εκφράσουν την ευγνωμοσύνη που νιώθω προς το πρόσωπο του Δρ. Ηλία Μαγλογιάννη, καθηγητή του Τμήματος Ψηφιακών Συστημάτων του Πανεπιστημίου Πειραιώς και πρόεδρο της επιτροπής μου. Υπήρξε πολύ υπομονετικός, συνεργατικός και με καθοδήγησε καταλλήλως.

Επίσης, είμαι ευγνώμων στην οικογένειά μου, ειδικά στους γονείς μου και τους φίλους μου. Η πίστη τους προς το πρόσωπό μου, κατάφερε να αποτελέσει σημαντικό κίνητρο κατά τη διάρκεια εκπόνησης της διπλωματικής μου εργασίας.

ΠΕΡΙΛΗΨΗ

Τα αθλητικά οπτικά δεδομένα είναι ένα ευρύ πεδίο στην τεχνητή νοημοσύνη και αποτελούν μέρος της καθημερινότητάς μας. Κάθε άθλημα έχει τους δικούς του κανόνες και τη δική του φιλοσοφία, με αποτέλεσμα οι αποφάσεις να ποικίλλουν και να διαφοροποιούνται αναλόγως την περίσταση. Δυστυχώς, οι αλγόριθμοι της τεχνητής νοημοσύνης δεν μπορούν να εφαρμοστούν σε κάθε περίπτωση, η οποία αντιστοιχεί σε πραγματικό σενάριο διαδραματιζόμενο στον καθημερινό κόσμο, το οποίο φυσικά απαιτεί συγκεκριμένες κινήσεις.

Θα πρέπει να είμαστε σε θέση να μοντελοποιήσουμε το πρόβλημα ούτως ώστε να οδηγηθούμε σε λογικά και αξιόλογα συμπεράσματα, τα οποία θα μας διευκολύνουν στη λήψη αποφάσεων. Κατά την ανάλυση ποδοσφαιρικών οπτικών δεδομένων έχει εντοπιστεί ανάγκη να υλοποιηθεί αλγόριθμος για την αξιοποίηση από ειδικούς του ποδοσφαίρου, αναλυτές και προπονητές. Έτσι, η ιδέα δημιουργίας ενός μοντέλου τεχνητής νοημοσύνης, εντοπίζεται να είναι αναγκαία, το οποίο αναγνωρίζει μέσω αποφάσεων μηχανικής μάθησης εάν ο σχηματισμός είναι ο συγκεκριμένος που έχουμε επιλέξει. Στο δικό μας σενάριο τα μοντέλα αρχικά αναγνωρίζουν τους ποδοσφαιριστές της δικής μας ομάδας και έπειτα παίρνουν απόφαση εάν η διάταξη είναι 4-4-2 ή δεν είναι.

Λέξεις Κλειδιά: Αναλυτική Ποδοσφαίρου, Μηχανική Μάθηση, Νευρωνικά Δίκτυα, Υπολογιστική Όραση.

ABSTRACT

Sport analytics is a broad field in artificial intelligence and is a part of our daily life. Each sport has its own rules and philosophy so decisions vary depending on the occasion. Also the AI algorithms cannot be applied to every scenario I come from the real world of sports because the decisions have to be specific. We should be able to vectorize the problem in order to arrive at conclusions that have decision value. In the analysis of football visual data a need has been identified to implement an algorithm for use by football experts, analysts and coaches. The idea of creating an artificial intelligence model that recognizes through machine learning decisions if the formation is the particular one we have chosen. In our scenario the models first identify the footballers of our own team and then make a decision whether the formation is 4-4-2 or not.

Keywords: Football analytics, Machine Learning, Neural Networks, Computer Vision.

ΠΕΡΙΕΧΟΜΕΝΑ

Κατάλογος Εικόνων.....viii

Κατάλογος Συντομογραφιών.....ix

Κεφάλαιο 1^ο - Εισαγωγή

1.1 Εισαγωγή.....1

Κεφάλαιο 2 - Σχετικές Εργασίες

2.1 Νευρωνικό Δίκτυο.....3

2.2 Συνελκτικά Νευρωνικά Δίκτυα.....4

2.3 Κατάτμηση Εικόνων.....8

2.4 Ανίχνευση Αντικειμένων με CNN.....9

2.4.1 Ανίχνευση Παίκτη.....9

2.4.2 Ανίχνευση Προσομοιωμένων Παικτών.....12

2.4.3 Αναγνώριση Αριθμού Φανέλας.....12

2.5 Ανάλυση Στάσης Αθλητικής Κάμερας.....13

2.5.1 Βαθμονόμηση Αθλητικής Κάμερας.....13

2.6 Τακτική Ανάλυση.....15

2.6.1 Εξαγωγή Στατιστικών Δεικτών.....15

2.6.2 Ταξινόμηση Τακτικού Σχηματισμού.....16

Κεφάλαιο 3 - Μεθοδολογία

Αρχιτεκτονική Μεθοδολογίας.....18

3.1 Σημασιολογική Τμηματοποίηση Παικτών.....19

3.1.1 Σύνολο Δεδομένων και Προεπεξεργασία Δεδομένων.....19

3.1.2 Μοντέλο Αρχιτεκτονικής U-Net με Μεταφερόμενη Γνώση.....	21
3.1.3 Μοντέλο Παραμετροποίησης και Μοντέλο Αξιολόγησης.....	28
3.1.4 Αποτελέσματα Ακρίβειας Μοντέλου.....	31
3.2 Αναγνώριση Προβολής Κάμερας.....	35
3.2.1 Σύνολο Δεδομένων και Προεπεξεργασία Δεδομένων.....	35
3.2.2 Επιλογή Μοντέλου.....	37
3.2.3 Αποτελέσματα Αξιολόγησης.....	38
3.3 Αναγνώριση Σχηματισμού 4-4-2.....	41
3.3.1 Σύνολο Δεδομένων και Προεπεξεργασία.....	41
3.3.2 Επιλογή Μοντέλου Τρίτου Σταδίου και Αποτελέσματα.....	42

Κεφάλαιο 4 - Πειράματα και Παρουσίαση

4.1 Επίδειξη Ανίχνευσης Παίκτη.....	44
4.2 Αναγνώριση Προβολής Κάμερας.....	46
4.3 Αναγνώριση Σχηματισμού 4-4-2.....	48

Κεφάλαιο 5 - Συζήτηση.....50

Κεφάλαιο 6 -Συμπεράσματα και Μελλοντικές Εργασίες.....52

Βιβλιογραφία.....53

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

1. Αρχιτεκτονική νευρωνικού δικτύου.....	3
2. Πίνακας A.....	5
3. Πίνακας B.....	5
4. Αποτέλεσμα συνέλιξης.....	7
5. Διαδικασία συνέλιξης.....	7
6. Μεθοδολογία εργασίας.....	18
7. Συντεταγμένες πολυγώνου.....	19
8. Μάσκα E.....	20
9. Δίκτυο U-Net.....	21
10. VGG16.....	23
11. Αρχιτεκτονική πυρήνα.....	24
12. Η διαδικασία τμηματοποίησης σύμφωνα με το UNet-VGG16 με μάθηση μεταφοράς.....	25
13. Μαθηματικός ορισμός της Διασταυρούμενης Εντροπίας. Σημειώστε ότι ο λογάριθμος υπολογίζεται στη βάση 2.....	29
14. ReLU v/s Σιγμοειδές.....	30
15. Παράδειγμα διασκελισμού.....	31
16. Η αναπλήρωση έχει οριστεί σε 1.....	32
17. Απώλεια ανά επανάληψη.....	34
18. Απώλεια επικύρωσης έναντι Απώλεια εκπαίδευσης.....	34
19. Μετατροπή της δομής εικόνας σε επίπεδο ισοπέδωσης.	36
20. Πίνακας με διανυσματικές εικόνες.	37
21. Έκθεση ταξινόμησης.....	40
22. Μάσκα συνόλου δεδομένων 3 ^{ου} αλγορίθμου.....	42
23. Αναφορά απόδοσης μοντέλου.....	43
24. Ολόκληρος ο σχηματισμός.....	44
25. Διαφορετική προβολή μάσκας.....	45
26. Μάσκα τερματοφύλακα.....	45
27. Δείγμα πανοραμικής προβολής 1.....	46

28. Δείγμα πανοραμικής προβολής 2.....	46
29. Δείγμα πίσω όψης 1.....	47
30. Δείγμα πίσω όψης 2.....	47
31. Σχηματισμός διαφορετικός	48
32. Διάταξη 4-4-2.....	49

ΚΑΤΑΛΟΓΟΣ ΣΥΝΤΟΜΟΓΡΑΦΙΩΝ

- ML: Machine learning
- JSON: JavaScript Object Notation
- CNN: Convolution Neural Network
- YOLO: You only look once
- SSD: Single Shot Detector

ΚΕΦΑΛΑΙΟ 1

ΕΙΣΑΓΩΓΗ

1.1 ΕΙΣΑΓΩΓΗ

Η πρώτη ενότητα αποτελείται από μια σύντομη εισαγωγή των αλγορίθμων μηχανικής μάθησης και πως εφαρμόζονται στο πεδίο της υπολογιστικής όρασης.

Η παρούσα εργασία αναλαμβάνει να εξετάσει ένα θέμα το οποίο ακούγεται αρκετά ενδιαφέρον στο ποδοσφαιρικό και εν γένει αθλητικό κοινό καθώς έρχεται να προτείνει μία ακολουθία μεθοδολογιών που μπορούν να εξάγουν χρήσιμα συμπεράσματα που αφορούν αποφάσεις τακτικού χαρακτήρα. Δύο σκέψεις οδήγησαν ως προς την ανάληψη οδήγησαν ως προς την ανάληψη αυτής της μελέτης, αρχικά οποιοσδήποτε στο μακρόκοσμο που γνωρίζουμε νιώθει την ανάγκη να εκφέρει άποψη και να προσπαθήσει να πείσει τους συνομιλητές του με επιχειρηματολογία για τις ποδοσφαιρικές και γενικά αθλητικές του γνώσεις. Έτσι σκεφτήκαμε γιατί να μην έχει και το δικαίωμα ένα μαθηματικό μοντέλο να μην μπορεί να εκφέρει την ίδια άποψη εξάγοντας αποτελέσματα που θα μπορούν να διαψεύσουν ή να επιβεβαιώσουν τη συζήτηση.

Εκτός από το πρώτο πρόβλημα που θα μπορούσε να εξετάσει η έρευνα μας ακολουθούσε και η ανάγκη που υπάρχει για εκσυγχρονισμό των προπονητικών αναλύσεων και επακόλουθων αποφάσεων με τη βοήθεια μοντέλων μηχανικής μάθησης. Είναι λογικό επακόλουθο λόγω του αριθμού όλων αυτών των αναλύσεων να επιλέξουμε ένα πρόβλημα να λύσουμε και αποφασίσαμε να δημιουργήσουμε μία λύση αναγνώρισης συγκεκριμένου σχηματισμού. Στην προκειμένη εργασία καταφέραμε να δημιουργήσουμε μία μεθοδολογία η οποία όμοια της δεν υπάρχει βάση βιβλιογραφικής προεπισκόπησης, δηλαδή να προσεγγίζεται το πρόβλημα της αναγνώρισης σχηματισμού με αλληλεπίδραση πληθώρας μαθηματικών μοντέλων.

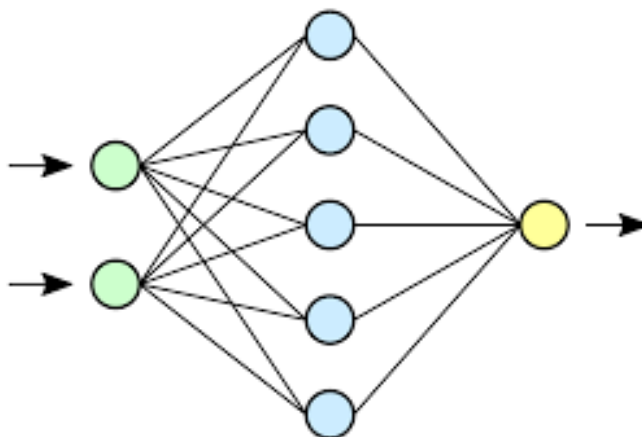
Τέλος, η δομή της έρευνας θα παρουσιαστεί πρώτα τμηματικά αναλύοντας κάθε αλγόριθμο ξεχωριστά και δεύτερον πειραματικά παρουσιάζοντας τα αποτελέσματα από τις δόκιμες και την άμεση εφαρμογή τους σε βιντεοληψίες.

ΚΕΦΑΛΑΙΟ 2

ΣΧΕΤΙΚΕΣ ΕΡΓΑΣΙΕΣ

2.1 ΝΕΥΡΩΝΙΚΟ ΔΙΚΤΥΟ

Τα τεχνητά νευρωνικά δίκτυα αποτελούνται από επίπεδα κόμβων, που περιέχουν ένα επίπεδο εισόδου, ένα ή περισσότερα επιστρώματα και ένα επίστρωμα εξόδου. Κάθε κόμβος, ή αλλιώς τεχνητός νευρώνας, συνδέεται με τον άλλο, έχοντας μία σχετική βαρύτητα και ένα οριακό κατώφλι. Εάν η έξοδος οποιουδήποτε μεμονωμένου κόμβου είναι πάνω από την καθορισμένη τιμή κατωφλίου, αυτός ο κόμβος ενεργοποιείται στέλνοντας δεδομένα στο επόμενο επίπεδο του δικτύου. Διαφορετικά, δεν μεταβιβάζονται δεδομένα στο επόμενο επίπεδο του δικτύου.



Εικόνα 1: Αρχιτεκτονική Νευρωνικού Δικτύου

Τα νευρωνικά δίκτυα βασίζονται σε δεδομένα εκπαίδευσης για να μάθουν και να βελτιώσουν την ακρίβειά τους με την πάροδο του χρόνου. Ωστόσο, μόλις αυτοί οι αλγόριθμοι μάθησης τελειοποιηθούν, τείνοντας σε ένα σημαντικό ποσοστό ακρίβειας, είναι ισχυρά εργαλεία της επιστήμης υπολογιστών και της τεχνητής νοημοσύνης, επιτρέποντάς μας να ταξινομούμε και να συγκεντρώνουμε δεδομένα με υψηλή ταχύτητα.

Οι εργασίες στην αναγνώριση ομιλίας ή την αναγνώριση εικόνας μπορεί να διαρκέσουν λεπτά έναντι ωρών σε σύγκριση με τη χειροκίνητη αναγνώριση από ειδικούς. Ένα από τα πιο γνωστά νευρωνικά δίκτυα είναι ο αλγόριθμος αναζήτησης της Google. Η μελέτη μας επικεντρώνεται στις τεχνικές Υπολογιστικής Όρασης και συγκεκριμένα σε αυτές που εφαρμόζονται στη σημασιολογική τμηματοποίηση των ποδοσφαιριστών μια ποδοσφαιρικής ομάδας με συγκεκριμένο χρώμα φανέλας.

2.2 ΣΥΝΕΛΙΚΤΙΚΑ ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Τα συνελικτικά νευρωνικά δίκτυα μπορούν να συλλάβουν με επιτυχία τις χωρικές και χρονικές εξαρτήσεις μιας εικόνας μέσω της εφαρμογής σχετικών φίλτρων. Η αρχιτεκτονική ταιριάζει καλύτερα στο σύνολο δεδομένων εικόνας λόγω της μείωσης του αριθμού των παραμέτρων που εμπλέκονται και της επαναχρησιμοποίησης των βαρύτητας. Με άλλα λόγια, το δίκτυο μπορεί να εκπαιδευτεί για να κατανοήσει καλύτερα την πολυπλοκότητα της εικόνας. Ο ρόλος αυτών των τύπων νευρωνικών δικτύων είναι να μειώσουν τις εικόνες σε μια μορφή που είναι ευκολότερη στην επεξεργασία, χωρίς να χάσουν χαρακτηριστικά που είναι κρίσιμα για τη λήψη μιας καλής πρόβλεψης. Αυτό είναι απαραίτητο όταν σχεδιάζουμε μια αρχιτεκτονική που δεν είναι μόνο καλή στην εκμάθηση χαρακτηριστικών, αλλά είναι επίσης επεκτάσιμη σε τεράστια σύνολα δεδομένων.

Μια συνέλιξη είναι ουσιαστικά ο πολλαπλασιασμός μεταξύ δύο πινάκων. Ο πολλαπλασιασμός αυτός είναι το βασικό συστατικό για την εκπαίδευση ενός

νευρωνικού δικτύου. Εκτελούμε πολλαπλασιασμό μήτρας όταν υπολογίζουμε κρυφές τιμές στρώματος που είναι ένας πολλαπλασιασμός μήτρας των τιμών εισόδου και των τιμών βάρους που συνδέουν την είσοδο με το κρυφό στρώμα. Ομοίως, εκτελούμε πολλαπλασιασμό μήτρας για να υπολογίσουμε τις τιμές του επιπέδου εξόδου. Για να εξηγήσουμε τη συνελικτική διαδικασία υποθέτουμε ότι έχουμε δύο πίνακες που μπορούμε να χρησιμοποιήσουμε ως παράδειγμα για να εκτελέσουμε συνέλιξη. Ακολουθούν οι πίνακες A και B:

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

Εικόνα 2: Πίνακας A

1	2
3	4

Εικόνα 3: Πίνακας B

Κατά την εκτέλεση της λειτουργίας συνέλιξης, σύρετε τον πίνακα B (τον μικρότερο πίνακα) πάνω από τον Πίνακα A (τον μεγαλύτερο πίνακα). Επιπλέον, εκτελούμε πολλαπλασιασμό στοιχείων προς στοιχείο μεταξύ του Πίνακα A και του Πίνακα B, ως εξής:

- Πολλαπλασιάστε $\{1,2,5,6\}$ του μεγαλύτερου πίνακα με $\{1,2,3,4\}$ του μικρότερου πίνακα:

$$1*1 + 2*2 + 5*3 + 6*4 = 44$$

- Πολλαπλασιάστε $\{2,3,6,7\}$ του μεγαλύτερου πίνακα με $\{1,2,3,4\}$ του μικρότερου πίνακα:

$$2*1 + 3*2 + 6*3 + 7*4 = 54$$

- Πολλαπλασιάστε $\{3,4,7,8\}$ του μεγαλύτερου πίνακα με $\{1,2,3,4\}$ του μικρότερου πίνακα:

$$3*1 + 4*2 + 7*3 + 8*4 = 64$$

- Πολλαπλασιάστε $\{5,6,9,10\}$ του μεγαλύτερου πίνακα με $\{1,2,3,4\}$ του μικρότερου πίνακα:

$$5*1 + 6*2 + 9*3 + 10*4 = 84$$

- Πολλαπλασιάστε $\{6,7,10,11\}$ του μεγαλύτερου πίνακα με $\{1,2,3,4\}$ του μικρότερου πίνακα:

$$6*1 + 7*2 + 10*3 + 11*4 = 94$$

- Πολλαπλασιάστε $\{7,8,11,12\}$ του μεγαλύτερου πίνακα με $\{1,2,3,4\}$ του μικρότερου πίνακα:

$$7*1 + 8*2 + 11*3 + 12*4 = 104$$

- Πολλαπλασιάστε $\{9,10,13,14\}$ του μεγαλύτερου πίνακα με $\{1,2,3,4\}$ του μικρότερου πίνακα:

$$9*1 + 10*2 + 13*3 + 14*4 = 124$$

- Πολλαπλασιάστε $\{10,11,14,15\}$ του μεγαλύτερου πίνακα με $\{1,2,3,4\}$ του μικρότερου πίνακα:

$$10*1 + 11*2 + 14*3 + 15*4 = 134$$

- Πολλαπλασιάστε $\{11,12,15,16\}$ του μεγαλύτερου πίνακα με $\{1,2,3,4\}$ του μικρότερου πίνακα:

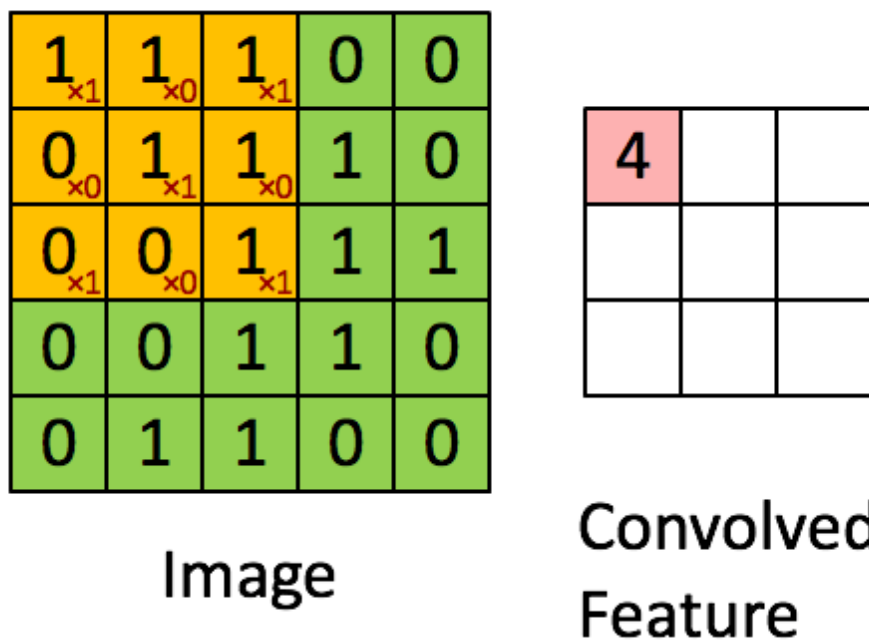
$$11*1 + 12*2 + 15*3 + 16*4 = 144$$

Το αποτέλεσμα της εκτέλεσης των προηγούμενων λειτουργιών φαίνεται ως εξής:

44	54	64
84	94	104
124	134	144

Εικόνα 4: Αποτέλεσμα συνέλιξης

Ο μικρότερος πίνακας ονομάζεται συνήθως φίλτρο ή πυρήνας, ενώ ο μεγαλύτερος πίνακας είναι η αρχική εικόνα.



Εικόνα 5: Διαδικασία Συνέλιξης

Σύμφωνα με τα αποτελέσματα, σημαντικό μέρος της συνελικτικής διαδικασίας είναι τα φίλτρα. Αυτά τα συστατικά είναι μήτρες αποτελούμενες από τιμές που ταξινομούνται τυχαία στην αρχή και στην περίπτωση που θέλουμε να πετύχουμε

συγκεκριμένο αποτέλεσμα θέτουμε κάποιες τιμές ώστε η βαρύτητα να είναι κατά κάποιον τρόπον προκατειλημμένη. Το μοντέλο μαθαίνει τις βέλτιστες τιμές βάρους ενός φίλτρου σε αυξανόμενες διεκπεραιώσεις.

Γενικά, όσο περισσότερα φίλτρα υπάρχουν στα συνελκτικά νευρωνικά δίκτυα, τόσο περισσότερα χαρακτηριστικά μιας εικόνας μπορεί να μάθει το μοντέλο. Όπως αναφέραμε προηγουμένως, κάθε φίλτρο διασχίζει την εικόνα ανά μία στήλη και ανά μία σειρά κάθε φορά (αφού εξαντλήσει όλες τις πιθανές στήλες μέχρι το τέλος της εικόνας). Αυτό είχε επίσης ως αποτέλεσμα το μέγεθος εξόδου να είναι 1 pixel μικρότερο από το μέγεθος της εικόνας εισόδου, τόσο από την άποψη του ύψους όσο και του πλάτους. Αυτό οδηγεί σε μερική απώλεια πληροφοριών και μπορεί να επηρεάσει τη δυνατότητα να προσθέσουμε την έξοδο της λειτουργίας συνέλιξης στην αρχική εικόνα.

2.3 ΚΑΤΑΤΜΗΣΗ ΕΙΚΟΝΩΝ

Στην ψηφιακή επεξεργασία εικόνας και την υπολογιστική όραση, η τμηματοποίηση εικόνας είναι η διαδικασία διαχωρισμού μιας ψηφιακής εικόνας σε πολλαπλά τμήματα, γνωστά και ως περιοχές εικόνας ή αντικείμενα εικόνας (σύνολα εικονοστοιχείων). Ο στόχος της τμηματοποίησης είναι να απλοποιήσει ή/και να αλλάξει την αναπαράσταση μιας εικόνας σε κάτι που είναι πιο ουσιαστικό και πιο εύκολο να αναλυθεί. [1],[2] Η τμηματοποίηση εικόνας χρησιμοποιείται συνήθως για τον εντοπισμό αντικειμένων και ορίων (γραμμές, καμπύλες κ.λπ.) σε εικόνες.

Πιο συγκεκριμένα, η τμηματοποίηση εικόνας είναι η διαδικασία εκχώρησης μιας ετικέτας σε κάθε εικονοστοιχείο μιας εικόνας, έτσι ώστε τα εικονοστοιχεία

με την ίδια ετικέτα να μοιράζονται ορισμένα χαρακτηριστικά. Το αποτέλεσμα της τμηματοποίησης της εικόνας είναι ένα σύνολο τμημάτων που καλύπτουν συλλογικά ολόκληρη την εικόνα ή ένα σύνολο περιγραμμάτων που εξάγονται από την εικόνα. Κάθε ένα από τα pixel σε μια περιοχή είναι παρόμοιο σε σχέση με κάποια χαρακτηριστική ή υπολογισμένη ιδιότητα [3], όπως το χρώμα, η ένταση ή η υφή. Οι

παρακείμενες περιοχές διαφέρουν σημαντικά ως προς το χρώμα σε σχέση με το ίδιο χαρακτηριστικό.

Η τμηματοποίηση εικόνας (image segmentation) έχει ομάδες υποκατηγοριών που ονομάζονται σημασιολογική τμηματοποίηση (semantic segmentation) και παν οπτική τμηματοποίηση [1]. [24] Η τμηματοποίηση στιγμιότυπων είναι μια προσέγγιση που προσδιορίζει, για κάθε pixel, μια παρουσία του αντικειμένου που ανήκει. Ανιχνεύει κάθε ξεχωριστό αντικείμενο ενδιαφέροντος της εικόνας. [4] Για παράδειγμα, κάθε άτομο σε ένα σχήμα είναι τμηματοποιημένο ως μεμονωμένο αντικείμενο. Η τμηματοποίηση εικόνας συνδυάζει σημασιολογική τμηματοποίηση και τμηματοποίηση στιγμιότυπων. Όπως και η σημασιολογική τμηματοποίηση, η τμηματοποίηση εικόνας είναι μια προσέγγιση που προσδιορίζει για κάθε εικονοστοιχείο, την κατηγορία που ανήκει. Σε αντίθεση με τη σημασιολογική κατάτμηση, η κατάτμηση εικόνων διακρίνει διαφορετικές παρουσίες της ίδιας κλάσης.

Τέλος αναφέρεται η σημασιολογική τμηματοποίηση, η οποία είναι μια προσέγγιση που ανιχνεύει, για κάθε εικονοστοιχείο, την κατηγορία του αντικειμένου [5]. [6] Αυτή η προσέγγιση χρησιμοποιείται στο σενάριό μας επειδή δεν υπάρχει ανάγκη ομαδοποίησης των αντικειμένων, είναι όμως απαραίτητο να αναγνωρίσουμε το σύνολο με την ίδια ετικέτα με την μοντελοποίηση εικονοστοιχείων.

2.4 ΑΝΙΧΝΕΥΣΗ ΑΝΤΙΚΕΙΜΕΝΩΝ ΜΕ CNN

2.4.1 Ανίχνευση Παίκτη

Όσον αφορά την ανάλυση βίντεο ποδοσφαίρου, το πρώτο ασφαλές συμπέρασμα που μπορούμε να εξάγουμε είναι να έχουμε τη δυνατότητα να προσδιορίσουμε τη θέση των παικτών δίνοντας βαρύτητα στις αποφάσεις που μπορούμε να πάρουμε ως αναλυτές ή προπονητές. Έτσι, ο κύριος στόχος είναι ο πράκτορας να μπορεί να εντοπίσει πρώτα τους παίκτες. Ανάλογα με το στόχο αυτού του προβλήματος, επιλέγουμε το σχετικό έργο του Komorowski και της ομάδας του [7] ενώ προσθέτουμε τον αφηρημένο στόχο ως προς την ανίχνευση αντικειμένων στην ορολογία.

Σε αυτό το έγγραφο, οι συγγραφείς έχουν σκοπό να ανιχνεύσουν τον χάρτη εμπιστοσύνης των παικτών και τα κουτιά οριοθέτησης των παικτών. Ως αλγόριθμο ανίχνευσης αντικειμένων, χρησιμοποίησαν ανιχνευτή μονής βολής (SSD) και τον αλγόριθμο *‘κοιτάζω μόνο μία φορά’* (YOLO). Η κύρια πρόκληση σε αυτά τα θέματα ανίχνευσης παικτών είναι η δυσκολία αναγνώρισης των παικτών σε διαφορετικές κλίμακες, καθώς το στιγμιότυπο αλλάζει είτε στην ανάλυση, είτε στην απόσταση προβολής της κάμερας.

Η κύρια μέθοδος αναγνώρισης επιτυγχάνεται με την εφαρμογή μη μέγιστης καταστολής στον χάρτη εμπιστοσύνης του παίκτη και τη λήψη όλων των τοποθεσιών με την εμπιστοσύνη πάνω από το όριο. Μια εναλλακτική προσέγγιση που παρατηρούμε στο έγγραφο του Keyu Lu [8], για το βασικό πρόβλημα που αντιμετώπισε, ήταν η δυναμική κάμερα. Δημιούργησε διαδοχικά συνελκτικά νευρωνικά δίκτυα (CNN) για την ταξινόμηση παικτών και μη παικτών με μια στρατηγική διαστολής για ακριβή ανίχνευση.

Με αφορμή το νευρωνικό δίκτυο που παρουσιάστηκε στο παραπάνω έγγραφο το οποίο έχει ως χαρακτηριστικό ότι εκπαιδεύεται χρησιμοποιώντας κομμάτια εικόνας, εφαρμόζοντάς το απευθείας σε μια ολόκληρη εικόνα που θα δημιουργούσε μη ευθυγραμμισμένους χάρτες χαρακτηριστικών για πυρήνες λειτουργίας. Για να αντιμετωπίσει αυτό το ζήτημα, ανέπτυξε μια στρατηγική διαστολής. Η στρατηγική διαστολής στοχεύει στην ευθυγράμμιση των χαρτών των χαρακτηριστικών πριν και μετά τη συνέλιξη και τη συγκέντρωση επιπέδων στις δοκιμές. Ο προηγούμενος αλγόριθμος εφαρμόζεται όχι μόνο στο ποδόσφαιρο αλλά και στο μπάσκετ. Ως εκ τούτου, αποδείχθηκε πολύ ικανοποιητικός στους χάρτες χαρακτηριστικών μπάσκετ καθώς είναι περισσότερο περίπλοκοι από εκείνους του ποδοσφαίρου.

Εκτός από την προηγούμενη έρευνα, υπάρχει μια εργασία σχετικά με την χρήση πολυτροπικού και πολυοπτικού αλγορίθμου, με σκοπό την ανίχνευση παικτών σε πραγματικό χρόνο σε ένα γήπεδο ποδοσφαίρου [9]. Αφορά την παρακολούθηση της πληρότητας των δημόσιων αθλητικών εγκαταστάσεων, οι οποίες είναι απαραίτητη για την αξιολόγηση της χρήσης τους, καθώς και την παρακίνηση για κατασκευή των

αθλητικών εγκαταστάσεων σε νέους χώρους. Στην περίπτωση ενός γηπέδου ποδοσφαίρου, η περιοχή που πρέπει να καλυφθεί είναι μεγάλη, επομένως πρέπει να χρησιμοποιηθούν αρκετές κανονικές κάμερες, γεγονός που καθιστά τη ρύθμιση δαπανηρή και περίπλοκη.

Ως εναλλακτική λύση, ανέπτυξαν ένα σύστημα που ανιχνεύει παίκτες από μια μοναδική φθηνή και ευρυγώνια κάμερα fisheye που υποστηρίζεται από μια ενιαία θερμική κάμερα στενής γωνίας. Σε αυτή την εργασία, υλοποιήθηκε ένα δίκτυο σε μια προσέγγιση εκχώρησης γνώσης στην οποία ο πράκτορας που εκπαιδεύεται και ο πράκτορας που εκπαιδεύει έχουν διαφορετικό τρόπο και

διαφορετική άποψη για την ίδια σκηνή. Σχεδιάζουν μια προσαρμοσμένη αύξηση εισχώρησης δεδομένων σε συνδυασμό με έναν αλγόριθμο ανίχνευσης κίνησης

για να χειριστούν την εκπαίδευση στην περιοχή της κάμερας fisheye που δεν καλύπτεται από τη θερμική. Έδειξαν ότι η λύση τους είναι αποτελεσματική στην ανίχνευση παικτών σε ολόκληρο το γήπεδο που γυρίστηκε από την κάμερα fisheye. Το αξιολόγησαν ποσοτικά και ποιοτικά στην περίπτωση μιας διαδικτυακής προσπάθειας απεικόνισης, όπου ο εκπαιδευόμενος πράκτορας ανιχνεύει παίκτες σε πραγματικό χρόνο ενώ προσαρμόζεται συνεχώς στις συνθήκες που ανιχνεύονται στη βιντεοσκόπηση.

Τέλος, όσον αφορά θέματα ανίχνευσης παικτών παρουσιάζεται μια προσέγγιση που δεν χρησιμοποιεί τα κλασικά νευρωνικά δίκτυα όπως χρησιμοποιούνται στην υπολογιστική όραση και την ανίχνευση αντικειμένων, αλλά χρησιμοποιεί γραμμικούς αλγόριθμους [14].

Σε αυτή την εργασία, αναλύθηκε μια μέθοδος ανίχνευσης αθλητικών παικτών που βασίζεται σε συνελκτικά νευρωνικά δίκτυα και συγκρίνεται με τη γνωστή προσέγγιση Adaboost που είναι γνωστό ότι αποδίδει καλά σε άλλα πλαίσια και άλλες περιπτώσεις. Σε αντίθεση με τις παραδοσιακές μεθόδους, αυτοί οι αλγόριθμοι δεν χρησιμοποιούν αφαίρεση φόντου μετά την εκτίμηση της κίνησης της κάμερας. Τα πειράματα έδειξαν

ότι και οι δύο μέθοδοι είναι πολύ ισχυρές και αποτελεσματικές. Συγκεκριμένα, τα συνελκτικά νευρωνικά δίκτυα αποδίδουν αποδεικνύονται πιο αποτελεσματικά, με μέσο δείκτη απόδοσης 4% σε διάφορα σύνολα δεδομένων.

2.4.2 Ανίχνευση Προσομοιωμένων Παικτών

Παρόμοια έρευνα έχει υλοποιηθεί για την ανίχνευση ρομπότ σε προσομοιωμένα παιχνίδια ποδοσφαίρου [10]. Στόχος της παρούσας εργασίας είναι να αναλύσει το γενικό πρόβλημα της χρήσης συνελκτικών νευρωνικών δικτύων (CNNs) σε ρομπότ με περιορισμένες υπολογιστικές δυνατότητες και να προτείνει γενικές κατευθυντήριες γραμμές σχεδιασμού για τη χρήση τους.

Επιπλέον, προτείνονται δύο διαφορετικοί ανιχνευτές CNN συνελκτικών νευρωνικών δικτύων σε ρομπότ NAO που μπορούν να λειτουργούν σε πραγματικό χρόνο ενώ παίζουν ποδόσφαιρο. Ένας από τους ανιχνευτές βασίζεται στο XNOR-Net [11] και ο άλλος στο SqueezeNet [12]. Κάθε ανιχνευτής μπορεί να επεξεργαστεί ένα αντικείμενο ρομπότ σε ~1ms, με μέσο αριθμό ανά 1,5 στιγμιότυπου που λαμβάνεται από την ανώτερη κάμερα του NAO. Το ποσοστό ανίχνευσης απόδοσης που λαμβάνεται είναι περίπου ίσο με 97%.

2.4.3 Αναγνώριση αριθμού φανέλας

Επίσης, μια ενδιαφέρουσα προσέγγιση που έχει παρόμοιες μεθόδους με την έρευνά μας είναι η αναγνώριση αριθμού φανέλας με CNNs. [13]. Σε αυτήν την έρευνα, παρουσιάζεται μια βαθιά συνελκτική προσέγγιση που βασίζεται σε νευρωνικά δίκτυα και προσεγγίζει το πρόβλημα της αυτόματης αναγνώρισης αριθμών φανέλας από βιντεοσκόπηση ποδοσφαίρου. Παρουσιάζονται δύο διαφορετικά σχήματα διανυσματικής κωδικοποίησης αριθμών φανέλας και συγκρίνονται μεταξύ τους. Ο πρώτος αντιμετωπίζει κάθε αριθμό ως ξεχωριστή κλάση, ενώ ο δεύτερος αντιμετωπίζει κάθε ψηφίο ως κλάση.

Επιπλέον, υλοποιείται μια ημιαυτόματη διαδικασία για την ανάλυση ενός συνόλου δεδομένων αριθμού φανέλας που αποτελείται από 8281 αριθμούς φανέλας. Ο καλύτερος ρυθμός αναγνώρισης 0,83 επιτεύχθηκε με την προτεινόμενη προσέγγιση με αύξηση δεδομένων και χωρίς τη χρήση εγκατάλειψης (dropout), σε σύγκριση με το 0,4 για ένα πιο παραδοσιακό ιστόγραμμα προσανατολισμένων κλίσεων (HOG) και διανυσματικής μηχανής υποστήριξης (SVM).

2.5 ΑΝΑΛΥΣΗ ΣΤΑΣΗΣ ΑΘΛΗΤΙΚΗΣ ΚΑΜΕΡΑΣ

Η εργασία μας χωρίζεται σε πολλαπλούς τομείς. Στον αντίποδα αναφέραμε σχετικές εργασίες σχετικά με το πρώτο μέρος της ανίχνευσης παικτών και τα άλλα έγγραφα ανίχνευσης αντικειμένων. Αυτό το μέρος θα παρουσιάσει έγγραφα όπου η βαθμονόμηση της κάμερας και οι προβολές είναι το κύριο ενδιαφέρον. Η προσέγγιση της εφαρμογής απαιτεί μια μέθοδο που μπορεί να αναγνωρίσει τη στάση της κάμερας και να πάρει την απόφαση στην ανάλυση.

2.5.1 Βαθμονόμηση Αθλητικής Κάμερας

Από όσα έχουν αναφερθεί προηγουμένως, οι ανάγκες που έχουν εμφανιστεί για αθλητικές κάμερες έχουν προτείνει και αναλύσει μια εξαιρετική αυτόματη μέθοδο βαθμονόμησης αθλητικών καμερών από μία και μόνο εικόνα χρησιμοποιώντας συνθετικά δεδομένα [15]. Αρχικά, ανέπτυξαν μια νέα μηχανή οπτικής κάμερας λεγόμενη ως πόζα. Η οπτική αυτής κάμερας έχει μόνο τρεις σημαντικές παραμέτρους, ώστε να μπορεί να παράγει αποτελεσματικά πολλές στάσεις κάμερας και αντίστοιχες εικόνες άκρων.

Στη συνέχεια, εκπαιδεύτηκαν με συμπαγή βαθιά χαρακτηριστικά μέσω ενός σιαμαίου δικτύου ξεκινώντας από ζευγαρωμένη εικόνα άκρης και ενώνοντας την πόζα κάμερας με αποτέλεσμα να έχουν δημιουργήσει μια βάση δεδομένων χαρακτηριστικών-

πόζας. Μετά από αυτό, χρησιμοποιούν ένα νέο μοντέλο διπλού-GAN για την ανίχνευση σημάτων πεδίου σε πραγματικές εικόνες.

Τέλος, αναζητούν μια αρχική πόζα κάμερας από τη βάση δεδομένων χαρακτηριστικών και βελτιώνουν τις πόζες της κάμερας χρησιμοποιώντας περικομμένες εικόνες εξ αποστάσεως. Αξιολογούμε τη μεθοδό μας τόσο σε συνθετικά όσο και σε πραγματικά δεδομένα. Η μεθοδός μας δεν αποδεικνύει μόνο την ανθεκτικότητα στα συνθετικά δεδομένα, αλλά επιτυγχάνει επίσης την ακρίβεια αιχμής

σε ένα τυπικό σύνολο δεδομένων ποδοσφαίρου και μια πολύ υψηλή απόδοση σε ένα σύνολο δεδομένων βόλεϊ.

Μέρος του προβλήματος που αναλύεται και στη συγκεκριμένη έρευνα είναι η ανάγκη κατανόησης του ποδοσφαιρικής βιντεοσκόπησης μολονότι έχει τραβήξει την προσοχή μεγάλης επιστημονικής κοινότητας μέσα σε λίγα χρόνια. Στην έρευνα που έγινε αντικείμενο μελέτης επικεντρώνεται στο θέμα της βαθμονόμησης κάμερας ως προς στους υφιστάμενους περιορισμούς της για την επιστημονική κοινότητα [16]. Πιο συγκεκριμένα, αντιμετώπισαν την απουσία ενός συνόλου δεδομένων βαθμονόμησης μεγάλης κλίμακας και ενός νευρωνικού δικτύου βαθμονόμησης προεκπαιδευμένο σε ένα τέτοιο σύνολο δεδομένων. Ενσωμάτωσαν ένα ισχυρό εμπορικό εργαλείο βαθμονόμησης σε μια πρόσφατη αρχιτεκτονική νευρωνικού δικτύου στο μεγάλης κλίμακας σύνολο δεδομένων SoccerNet, που αποτελείται από βίντεο μετάδοσης 500 ποδοσφαιρικών αγώνων.

Κυκλοφόρησαν το συγκεκριμένο δίκτυό τους και το αξιοποίησαν για να παρέχουν 3 τρόπους αναπαράστασης των αποτελεσμάτων βαθμονόμησης μαζί με τον εντοπισμό παικτών. Τέλος, αξιοποίησαν αυτές τις αναπαραστάσεις στο πλαίσιο της τρέχουσας αρχιτεκτονικής για το έργο εντοπισμού δράσης του SoccerNetv2 και πέτυχαν νέες υπερσύγχρονες επιδόσεις.

2.6 ΤΑΚΤΙΚΗ ΑΝΑΛΥΣΗ

Το τελευταίο μέρος της εργασίας περιέχει πορίσματα που εξάχθηκαν από την τακτική αθλητική ανάλυση χρησιμοποιώντας μοντέλα μηχανικής μάθησης για την αναγνώριση σχηματισμού..

2.6.1 Εξαγωγή Στατιστικών Δεικτών

Ο κόσμος του αθλητισμού περιλαμβάνει εγγενώς γρήγορα και πολύπλοκα γεγονότα που είναι δύσκολο για τους προπονητές και τους αναλυτές να επεξεργαστούν και να

αναλύσουν και για το κοινό να ακολουθήσει. Σε ομαδικά αθλήματα με γρήγορο ρυθμό, όπως το ποδόσφαιρο, η παρακολούθηση όλων των παικτών και η ανάλυση της απόδοσής τους μετά από κάθε αγώνα είναι πολύ δύσκολη. Τα τρέχοντα σενάρια για τον εντοπισμό των καλύτερων ταλέντων στο ποδόσφαιρο περιλαμβάνουν από στόμα σε στόμα και προπονητές / υπεύθυνους προσλήψεων που ψάχνουν ώρες βίντεο με μη αυτόματο σχολιασμό. Πρόκειται για μια πολύ δαπανηρή και επίπονη διαδικασία και προκατειλημμένη από τη φύση της λήψης των αποφάσεων. Για να αμβλυθούν αυτά τα προβλήματα, η προσκείμενη εργασία [17] προτείνει ένα αυτοματοποιημένο σύστημα που μπορεί να ανιχνεύσει, να παρακολουθήσει και να ταξινομήσει τις ομάδες πολλών παικτών και να αναγνωρίσει τον παίκτη που ελέγχει την μπάλα σε ένα βίντεο.

Το σύστημα παράγει τρία πολύ σημαντικά στατιστικά τακτικής για έναν παίκτη:

- 1) διάρκεια κατοχής μπάλας
- 2) αριθμός επιτυχημένων πασών
- 3) αριθμός επιτυχημένων κλεψιμάτων

Αυτό γίνεται με την εκπαίδευση συνελκτικών νευρωνικών δικτύων (CNNs) για:

- (α) τον εντοπισμό και την παρακολούθηση των παικτών στο γήπεδο
- (β) την ταξινόμηση της ομάδας ενός ανιχνευμένου παίκτη
- (γ) την αναγνώριση του παίκτη που ελέγχει την μπάλα
- (δ) τη συγκέντρωση όλων των πληροφοριών που εξάγονται από τα (α), (β) και (γ) για τη δημιουργία των στατιστικών των παικτών.

Για να ξεπεραστεί το πρόβλημα, τα χαρακτηριστικά που εξάγονται από συγκεκριμένους αγώνες ποδοσφαίρου δεν γενικεύονται απαραίτητα σε διαφορετικούς ποδοσφαιρικούς αγώνες, το έγγραφο προτείνει ελάχιστο αριθμό σχολιασμού και αύξησης δεδομένων για συγκεκριμένους αγώνες, χρησιμοποιώντας μια παραλλαγή των Deep Convolutional Generative Adversarial Networks (DCGAN) για τη βελτίωση της ακρίβειας. Τα πειραματικά αποτελέσματα και οι πειραματικές μελέτες δείχνουν ότι η προτεινόμενη προσέγγιση ξεπερνά τις σύγχρονες προσεγγίσεις όσον αφορά την ακρίβεια και την ταχύτητα επεξεργασίας.

2.6.2 Ταξινόμηση Τακτικού Σχηματισμού

Το τελικό πρόβλημα που επιλύεται και έχει ως σκοπό η συγκεκριμένη εργασία αφορά το θέμα της ταξινόμησης σχηματισμού. Βιβλιογραφικά στην ακόλουθη έρευνα έχουν δημιουργηθεί αυτοματοποιημένα εργαλεία για την υποστήριξη της απαιτούμενης διαδικασίας ανάλυσης. Σε αυτό το πλαίσιο, ένα από τα κύρια καθήκοντα ήταν να συνοψίσουμε τους σχηματισμούς ομάδων με μοτίβα όπως το 4-4-2. Στη παρούσα βιβλιογραφική έρευνα [18], εισήχθει μια προσέγγιση ανάλυσης που ταξινομεί και οπτικοποιεί αυτόματα τον σχηματισμό της ομάδας με βάση τα δεδομένα θέσης των παικτών. Επικεντρώθηκαν σε καταστάσεις ενός αγώνα αντί για ολοκληρωμένους χρόνους ημιχρόνου ή αγώνων για να παρέχουν μια πιο λεπτομερή ανάλυση.

Η νέα προσέγγιση ταξινόμησης υπολογίζει την ομοιότητα με βάση προκαθορισμένα πρότυπα για διαφορετικούς τακτικούς σχηματισμούς. Παρέχεται λεπτομερής ανάλυση

των μεμονωμένων καταστάσεων αγώνα ανάλογα με την κατοχή μπάλας και το μήκος του τμήματος αγώνα. Για το σκοπό αυτό, χρησιμοποιήθηκε μια συμπερίληψη εικόνων που συνοψίζει το σχηματισμό της ομάδας σε ένα τμήμα αγώνα, η προηγούμενη μελέτη που καταδεικνύει:

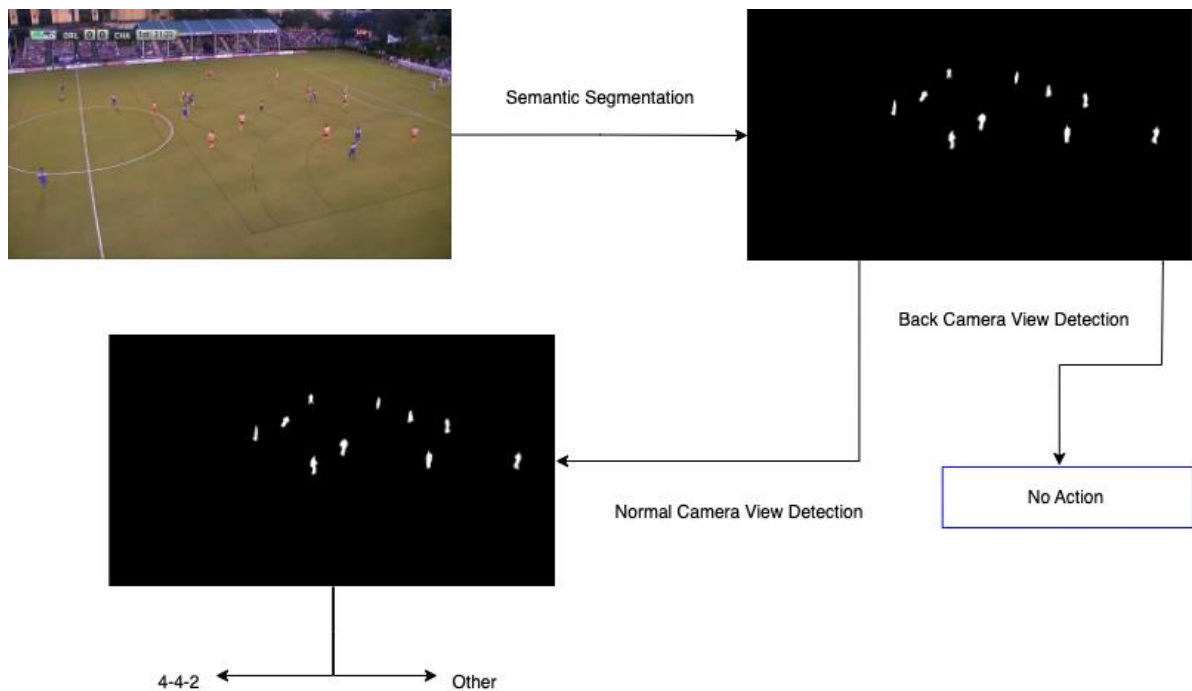
- 1) την πολυπλοκότητα της εργασίας
- 2) τη χρησιμότητα της απεικόνισης μεμονωμένων καταστάσεων για την κατανόηση των σχηματισμών της ομάδας.

Η προτεινόμενη προσέγγιση ταξινόμησης υπερτερεί των υφιστάμενων μεθόδων ταξινόμησης σχηματισμού. Ο συγκεκριμένος αλγόριθμος τους έδωσε πληροφορίες σχετικά με τις αδυναμίες της χρήσης μοτίβων όπως το 4-4-2 για την περιγραφή των σχηματισμών ομάδων.

ΚΕΦΑΛΑΙΟ 3

ΜΕΘΟΔΟΛΟΓΙΑ

Η ιδέα πίσω από την συγκεκριμένη έρευνα ήταν η υλοποίησης μιας λύσης που να είναι αυτόνομη και να μπορεί να μεταποιηθεί σε εμπορικό προϊόν. Βάση αυτής της υπόθεσης ο αλγόριθμος ικανοποιεί όλες τις φάσεις της λήψης αποφάσεων αναγνώρισης σχηματισμού. Το πρώτο συστατικό αυτού του αγωγού είναι η τμηματοποίηση των παικτών. Το δεύτερο είναι η ανίχνευση της κάμερας από άποψη λήψης και τρίτον η αναγνώριση ενός συγκεκριμένου επιθυμητού σχηματισμού. Οι παρακάτω εικόνες απεικονίζουν τη λύση που προτείνουμε σε αυτή την διατριβή και όλες τις λειτουργίες βημάτων που συνδέουν όλες αυτές τις αποφάσεις αλγορίθμων. Η λύση, όπως θα αναλύσουμε παρακάτω, εξάγει άκρως ενδιαφέροντα αποτελέσματα ως προς την υπολογιστική όραση και την ποικιλότητα στις λύσεις.



Εικόνα 6: Μεθοδολογία εργασίας.

ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΜΕΘΟΔΟΛΟΓΙΑΣ

3.1 ΣΗΜΑΣΙΟΛΟΓΙΚΗ ΤΜΗΜΑΤΟΠΟΙΗΣΗ ΠΑΙΚΤΩΝ

Όπως αναφέραμε το πρώτο βήμα αυτής της μεθοδολογίας είναι η αναγνώριση των παικτών της ομάδας μας. Η δομή της τρέχουσας αρχιτεκτονικής αποτελείται από την αποτύπωση του περιεχομένου και από μία συμμετρική επεκτεινόμενη μάσκα που εντοπίζει με ακρίβεια τους παίκτες μας.

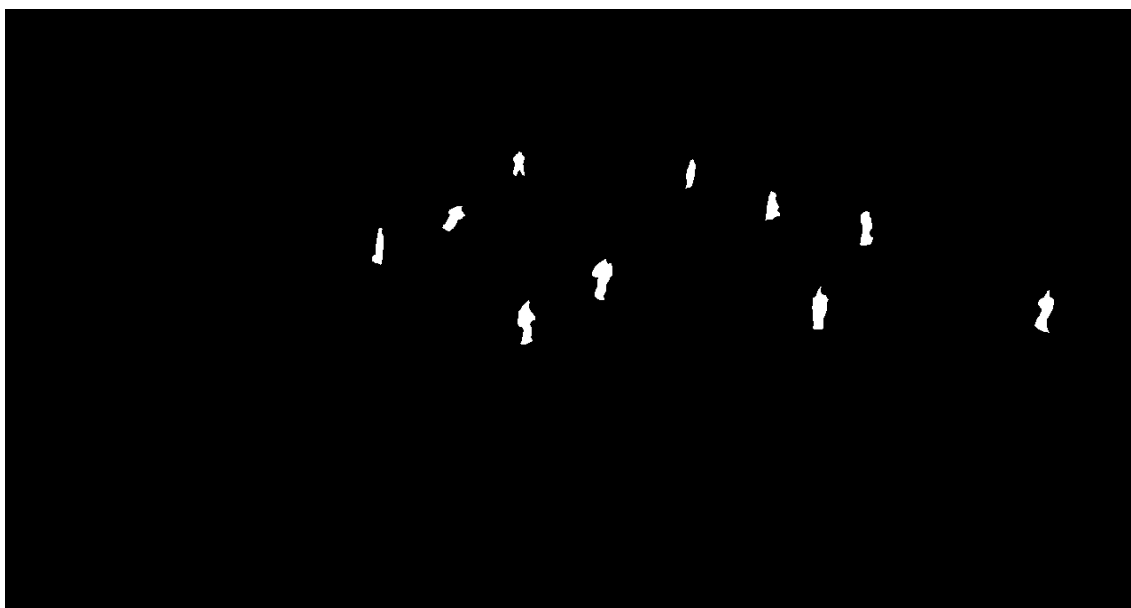
3.1.1 Σύνολο Δεδομένων και Προεπεξεργασία Δεδομένων

Η τρέχουσα ανάγκη μιας ολοκληρωμένης λύσης έθεσε ως αρχή τη λήψη αποφάσεων κατά τη διάρκεια ροής βίντεο. Ως αποτέλεσμα, λάβαμε υπόψιν για το ακόλουθο σύνολο δεδομένων που έχει χρησιμοποιηθεί στην τρέχουσα εργασία [8]. Οι απαιτήσεις του προβλήματος που προσπαθούμε να επιλύσουμε δεν απαιτούν τη χρήση των πλαισίων οριοθέτησης ως προς το συγκεκριμένο σύνολο δεδομένων που χρησιμοποιείται για μια άλλη υλοποίηση. Αντιθέτως απαιτεί τη χρήση πολυγώνων που εξάγονται γύρω από τις συντεταγμένες κάθε παίκτη με υψηλό επίπεδο λεπτομέρειας καταγεγραμμένη. Υπάρχουν επίσης διαφορετικές μορφές στις εξαγόμενες συντεταγμένες με βάση το πρόβλημα που προσπαθούμε να επιλύσουμε. Για παράδειγμα, για προβλήματα τμηματοποίησης καταγράφουμε σε μορφή JSON όλες τις συντεταγμένες που έχουν εξαχθεί με το χέρι. Η παρακάτω εικόνα παρουσιάζει τη διαδικασία για κάθε εικόνα.



Εικόνα 7: Συντεταγμένες Πολυγώνου.

Από την τρέχουσα διαδικασία είναι λογικό να συμπεράνουμε ότι είναι δύσκολο να εξαχθούν πολύγωνα σε σχεδόν δύο χιλιάδες εικόνες. Έτσι, μειώσαμε τον αριθμό των εικόνων από το αρχικό σύνολο δεδομένων ως σύνολο εκπαίδευσης σε 161 στοιχεία και το σύνολο δοκιμής σε 33 στοιχεία. Αυτή η διαδικασία ήταν η πιο χρονοβόρα κατά την διάρκεια της διαδικασίας της εργασίας. Το τελικό βήμα της προ επεξεργασίας πριν αρχίσουμε να ωθούμε δεδομένα σε τεχνικές μηχανικής μάθησης είναι να εξαγάγουμε τη μάσκα κάθε εικόνας. Αυτός ήταν ο λόγος για τον οποίο χρησιμοποιήσαμε πολύγωνο, διότι χρειαζόταν ο λεπτομερής εντοπισμός των παικτών για την εξαγωγή μασκών μεγάλης ακρίβειας. Αποτυπώνεται στις παρακάτω εικόνες η τελική μορφή του συνόλου δεδομένων εικόνας.



Εικόνα 8: Μάσκα E

3.1.2 Μοντέλο Αρχιτεκτονικής U-Net με Μεταφερόμενη Γνώση

Σκοπός του πρώτου βήματος της υλοποιήσας αρχιτεκτονικής είναι η δημιουργία ενός μαθηματικού μοντέλου που βασίζεται σε αλγόριθμο εντοπισμού θέσης των παικτών. Ο αλγόριθμος του U-Net [19] ανιχνεύει την κλάση που αντιστοιχεί σε όλα τα pixel της εικόνας. Το U-Net υβριδοποιείται με την αρχιτεκτονική VGG16 ως επίπεδο κωδικοποιητή [20] για την απλοποίηση της αρχιτεκτονικής και την αντιμετώπιση του προβλήματος της ταξινόμησης δεδομένων μικρού μεγέθους. Στον αντίποδα, η πολυπλοκότητα του U-Net συχνά ξοδεύει πολύ χρόνο στην εκτέλεσή του και επηρεάζεται σε μεγάλο βαθμό όταν ο υπολογιστικές προδιαγραφές είναι ανεπαρκείς. Λαμβάνοντας υπόψη αυτές τις παραμέτρους, εκτελούμε τη μεταφορά γνώσης στο μέρος των δεδομένων εκπαίδευσης στο υβρίδιο U-Net και VGG16. Το αναφερόμενο μοντέλο εξετάζει τη χαρτογράφηση pixel-to-pixel και την διαδικασία ground-truth για να προσδιορίσει την κατηγορία pixel. Το επόμενο σχήμα παρουσιάζει το U-Net δίκτυο.

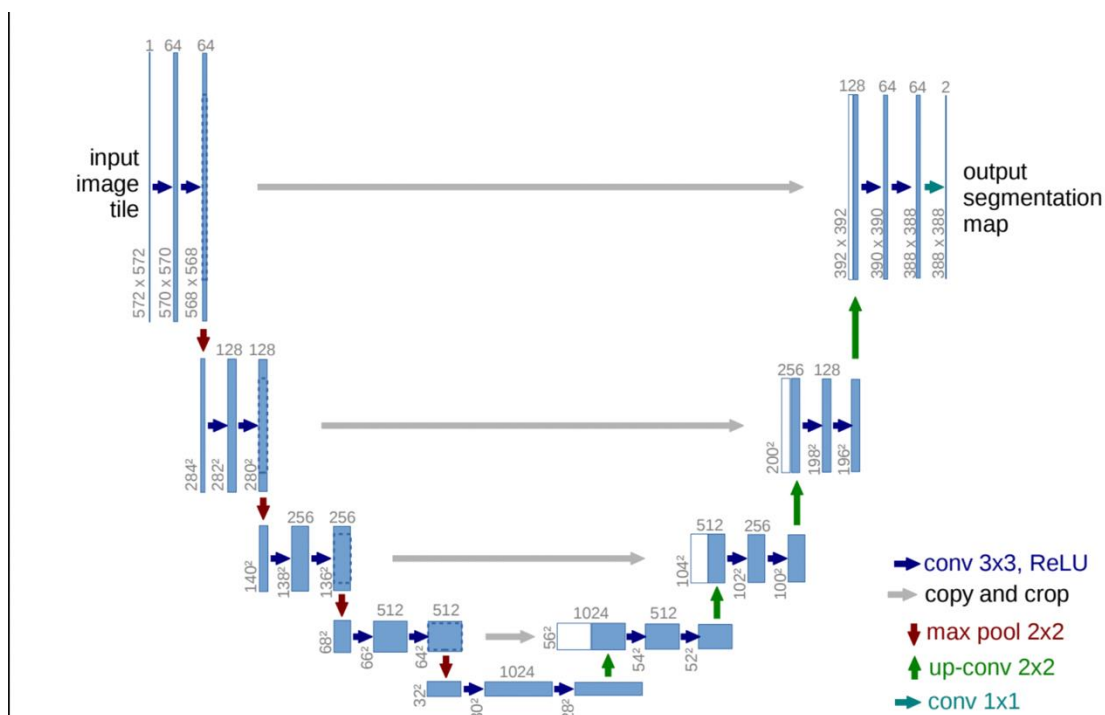


Fig. 1. U-net architecture (example for 32x32 pixels in the lowest resolution). Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. White boxes represent copied feature maps. The arrows denote the different operations.

Εικόνα 9: Δίκτυο U-Net.

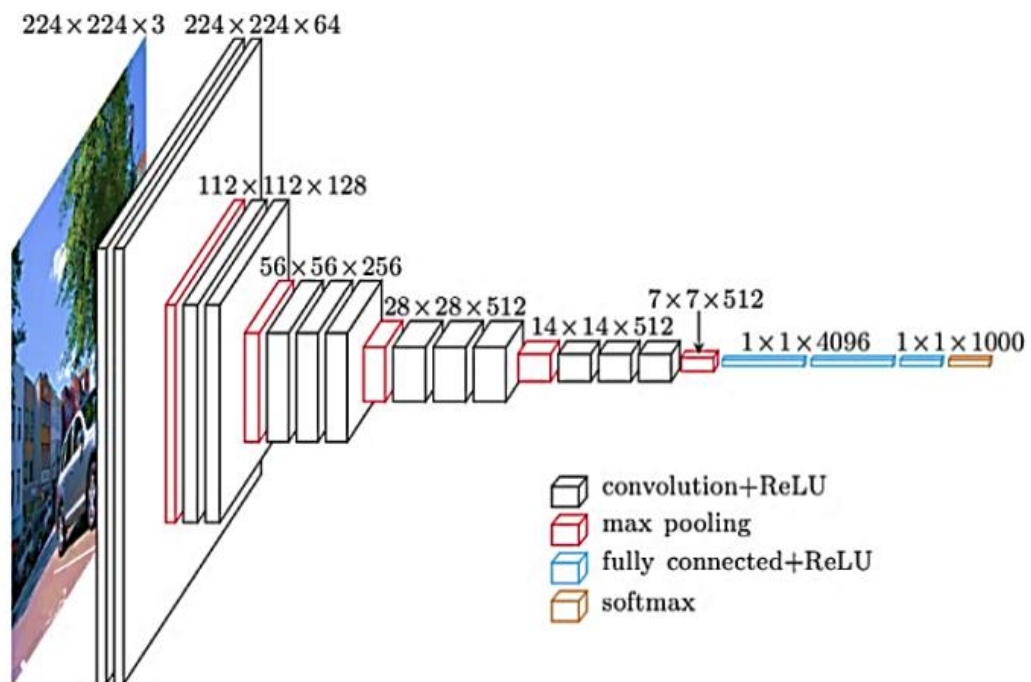
Το όνομα της αρχιτεκτονικής U-net προέρχεται από το σχήμα U που αναπαρίσταται κατά την μοντελοποίηση του. Το δίκτυο περιέχει δύο διαδρομές. Η αριστερή διαδρομή ονομάζεται κωδικοποιητής και η δεξιά διαδρομή αποκωδικοποιητής (επεκτεινόμενο επίπεδο). Αναλυτικότερα, ο κωδικοποιητής είναι ένα δίκτυο στο οποίο η έξοδος είναι ο χάρτης χαρακτηριστικών, δηλαδή ένα διάνυσμα που περιέχει τις πληροφορίες που αντιπροσωπεύουν την είσοδο. Ο αποκωδικοποιητής που έχει την ίδια δομή αλλά λειτουργεί με αντίθετο τρόπο, αναπαριστά ένα δίκτυο που λαμβάνει χάρτες χαρακτηριστικών από τον κωδικοποιητή και παρέχει μια παρόμοια αντιστοιχία της πραγματικής εισόδου ή της επιδιωκόμενης εξόδου.

Η διαδικασία στη διαδρομή του κωδικοποιητή μειώνει τον πίνακα εισόδου μεγέθους αυξάνοντας τον αριθμό των χαρτών χαρακτηριστικών, ενώ στη διαδρομή του αποκωδικοποιητή επιστρέφει ο πίνακας στο αρχικό του μέγεθος ελαχιστοποιώντας τον αριθμό των χαρτών χαρακτηριστικών. Επομένως, τα αποτελέσματα της τμηματοποίησης, μπορούν να συγκριθούν με την μεθοδολογία ground-truth σε κάθε εικονοστοιχείο.

Το προτεινόμενο μοντέλο μειώνει το επίπεδο διανυσμάτων και τις παράμετρους του U-Net συνδυάζοντάς το με μια άλλη αρχιτεκτονική, το VGG16. Η επιλογή του VGG16 οφείλεται στην ομοιότητά του με το συμβατικό στρώμα του U-Net και ο αριθμός των παραμέτρων είναι μικρότερος από το U-Net. Επιπλέον, το VGG16 έχει ήδη βάρη από παραμέτρους που είναι εύκολα προσβάσιμες, και έχει ως αποτέλεσμα αυτά τα βάρη να μπορούν να προσαρμοστούν σε αυτό το νέο μοντέλο. Αρκετά μοντέλα που χρησιμοποιούνται για τμηματοποίηση συχνά αποτελούνται από το στρώμα συστολής και το επίπεδο διαστολής.

Σε αυτήν τη μελέτη, τροποποιήσαμε το VGG16 ώστε να μοιάζει με την αρχιτεκτονική U-Net προσθέτοντας ένα εκτεταμένο στρώμα που αποτελείται από διάφορα δειγματοληπτικά επίπεδα, καθώς και στρώματα συνέλιξης στο τέλος της αρχιτεκτονικής VGG16. Αυτό επαναλαμβάνεται μέχρι η συνολική αρχιτεκτονική του μοντέλου να είναι συμμετρική και να μοιάζει με το σχήμα του γράμματος U. Επομένως, στην αρχιτεκτονική του μοντέλου UNet-VGG16, θα έχουμε ένα συμβατικό στρώμα, το οποίο είναι το VGG16, και το στρώμα επέκτασης που θα προστεθεί αργότερα. Με αυτή

τη νέα αρχιτεκτονική, οι παράμετροι θα μειωθούν σημαντικά. Επίσης η παρακάτω εικόνα αναπαριστά τη δομή του δικτύου VGG16 προκειμένου να εισαγάγουμε την ομοιότητα με την αρχιτεκτονική U-Net [19].



Εικόνα 10: VGG16

Όπως αναλύθηκε στις απαιτήσεις της αρχιτεκτονικής, το μέγεθος που απαιτείται για κάθε εικόνα είναι 224×224 , οπότε υπάρχει μετασχηματισμός δεδομένων και τροποποίηση εικόνας σχετικά με την εικόνα εισόδου στο νευρωνικό δίκτυο. Οι δύο πτυχές που πρέπει να μας απασχολούν κατά την εκτέλεση της τμηματοποίησης εικόνας είναι οι ακόλουθες:

1. Το σχήμα.
2. Η δομή.

Συγκεκριμένα, η δομή και το σχήμα των αντικειμένων στην αρχική εικόνα παραμένουν τα ίδια στην τμηματοποιημένη έξοδο και η αξιοποίηση μιας πλήρως συνελκτικής

αρχιτεκτονικής (και όχι μιας μορφής διαμόρφωσης επίστρωσης σε μία διάσταση) μπορεί να βοηθήσει καθώς χρησιμοποιούμε μια εικόνα ως είσοδο και μια άλλη ως έξοδο. Όπως αναφέρθηκε προηγουμένως, σχετικά με τον αποκωδικοποιητή, καθώς ενεργούν τα παραδοσιακά συνελκτικά επίπεδα, μειώνουν το μέγεθος της εικόνας ενώ αυξάνουν τον αριθμό των καναλιών.

Επιπλέον, κατά την αύξηση των επιπέδων στον αποκωδικοποιητή, αξιοποιούμε πληροφορίες από τα αντίστοιχα επίπεδα στο αριστερό μισό χρησιμοποιώντας συνδέσεις παράλειψης χαρακτηριστικών, ώστε να είμαστε σε θέση να διατηρήσουμε τη δομή και τα αντικείμενα στην αρχική εικόνα. Με τον συγκεκριμένο τρόπο, η αρχιτεκτονική U-Net μαθαίνει να διατηρεί τη δομή (και τα σχήματα των αντικειμένων) της αρχικής εικόνας, ενώ αξιοποιεί τα χαρακτηριστικά της συνέλιξης από τη μεταφορά γνώσης για να προβλέψει τις κλάσεις που αντιστοιχούν σε κάθε εικονοστοιχείο. Σε γενικές γραμμές, έχουμε τόσα κανάλια στην έξοδο όσος είναι ο αριθμός των κλάσεων που θέλουμε να προβλέψουμε και το σενάριο που εφαρμόζουμε ορίζει δύο κατηγορίες, τους παίκτες και το φόντο.

```
self.encoder = vgg16_bn(pretrained=pretrained).features
self.block1 = nn.Sequential(*self.encoder[:6])
self.block2 = nn.Sequential(*self.encoder[6:13])
self.block3 = nn.Sequential(*self.encoder[13:20])
self.block4 = nn.Sequential(*self.encoder[20:27])
self.block5 = nn.Sequential(*self.encoder[27:34])

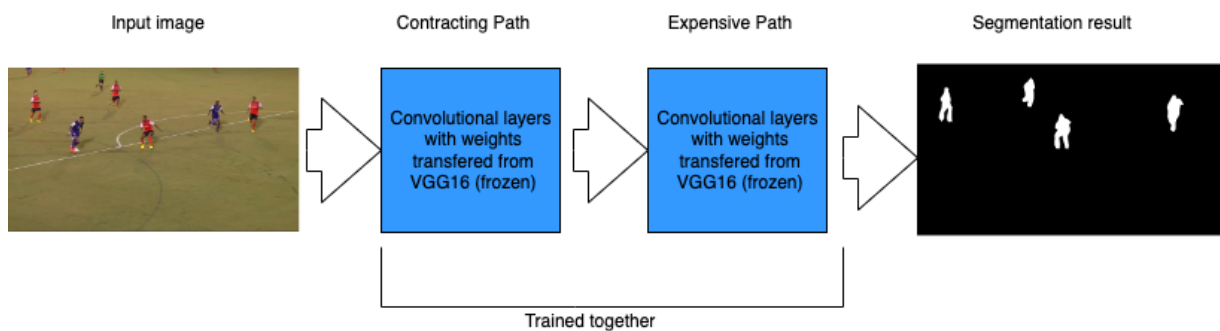
self.bottleneck = nn.Sequential(*self.encoder[34:])
self.conv_bottleneck = conv(512, 1024)

self.up_conv6 = up_conv(1024, 512)
self.conv6 = conv(512 + 512, 512)
self.up_conv7 = up_conv(512, 256)
self.conv7 = conv(256 + 512, 256)
self.up_conv8 = up_conv(256, 128)
self.conv8 = conv(128 + 256, 128)
self.up_conv9 = up_conv(128, 64)
self.conv9 = conv(64 + 128, 64)
self.up_conv10 = up_conv(64, 32)
self.conv10 = conv(32 + 64, 32)
self.conv11 = nn.Conv2d(32, out_channels, kernel_size=1)
```

Εικόνα 11: Αρχιτεκτονική πυρήνα.

Αναλύοντας και μελετώντας την μεθοδολογία του πρώτου τμήματος, η εφαρμογή σε αυτή το σημείο έχει την ακόλουθη δομή που αναπαρίσταται στο σχήμα 11, όπου τα πρώτα μπλοκ μέχρι το σημείο συμφόρησης περιέχουν τα εξαγόμενα χαρακτηριστικά από τα προεκπαιδευμένα βάρη δικτύου VGG16 τα αποκαλούμενα κωδικοποιητές και δεύτερον έρχεται ο αποκωδικοποιητής με τη ανάστροφη συνέλιξη και την αύξηση των επιστρωμάτων σε αύξουσα σειρά.

Στην εικόνα 12 παρουσιάζεται η διαδικασία μοντελοποίησης της τμηματοποίησης εικόνας σύμφωνα με τη νέα προτεινόμενη αρχιτεκτονική και ακολούθως θα αναλυθούν και τα υπόλοιπα κομμάτια της εναπομένουσας αρχιτεκτονικής.



Εικόνα 12: Η διαδικασία τμηματοποίησης σύμφωνα με το UNet-VGG16 με μάθηση μεταφοράς.


```

UNet(
  (encoder): Sequential(
    (0): Conv2d(3, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (1): BatchNorm2d(64, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (2): ReLU(inplace=True)
    (3): Conv2d(64, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (4): BatchNorm2d(64, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (5): ReLU(inplace=True)
    (6): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
    (7): Conv2d(64, 128, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (8): BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (9): ReLU(inplace=True)
    (10): Conv2d(128, 128, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (11): BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (12): ReLU(inplace=True)
    (13): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
    (14): Conv2d(128, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (15): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (16): ReLU(inplace=True)
    (17): Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (18): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (19): ReLU(inplace=True)
    (20): Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (21): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (22): ReLU(inplace=True)
    (23): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
    (24): Conv2d(256, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (25): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (26): ReLU(inplace=True)
    (27): Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (28): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (29): ReLU(inplace=True)
    (30): Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (31): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (32): ReLU(inplace=True)
    (33): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
    (34): Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (35): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (36): ReLU(inplace=True)
    (37): Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (38): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (39): ReLU(inplace=True)
    (40): Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (41): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (42): ReLU(inplace=True)
    (43): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False))
  (block1): Sequential(
    (0): Conv2d(3, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (1): BatchNorm2d(64, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (2): ReLU(inplace=True)
    (3): Conv2d(64, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (4): BatchNorm2d(64, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (5): ReLU(inplace=True))
  (block2): Sequential(
    (0): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
    (1): Conv2d(64, 128, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (2): BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (3): ReLU(inplace=True)
    (4): Conv2d(128, 128, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (5): BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (6): ReLU(inplace=True))
  (block3): Sequential(
    (0): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
    (1): Conv2d(128, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (2): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (3): ReLU(inplace=True)
    (4): Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (5): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (6): ReLU(inplace=True))
  (block4): Sequential(
    (0): Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (1): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (2): ReLU(inplace=True)
    (3): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
    (4): Conv2d(256, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (5): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (6): ReLU(inplace=True))
  (block5): Sequential(
    (0): Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))

```



```

(1): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
(2): ReLU(inplace=True)
(3): Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
(4): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
(5): ReLU(inplace=True)
(6): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
(bottleneck): Sequential(
  (0): Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
  (1): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (2): ReLU(inplace=True)
  (3): Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
  (4): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (5): ReLU(inplace=True)
  (6): Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
  (7): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (8): ReLU(inplace=True)
  (9): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
(conv_bottleneck): Sequential(
  (0): Conv2d(512, 1024, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
  (1): BatchNorm2d(1024, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (2): ReLU(inplace=True)
(up_conv6): Sequential(
  (0): ConvTranspose2d(1024, 512, kernel_size=(2, 2), stride=(2, 2))
  (1): ReLU(inplace=True)
(conv6): Sequential(
  (0): Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
  (1): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (2): ReLU(inplace=True)
(up_conv7): Sequential(
  (0): ConvTranspose2d(512, 256, kernel_size=(2, 2), stride=(2, 2))
  (1): ReLU(inplace=True)
(conv7): Sequential(
  (0): Conv2d(768, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
  (1): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (2): ReLU(inplace=True)
(up_conv8): Sequential(
  (0): ConvTranspose2d(256, 128, kernel_size=(2, 2), stride=(2, 2))
  (1): ReLU(inplace=True)
(conv8): Sequential(
  (0): Conv2d(384, 128, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
  (1): BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (2): ReLU(inplace=True)
(up_conv9): Sequential(
  (0): ConvTranspose2d(128, 64, kernel_size=(2, 2), stride=(2, 2))
  (1): ReLU(inplace=True)
(conv9): Sequential(
  (0): Conv2d(192, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
  (1): BatchNorm2d(64, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (2): ReLU(inplace=True)
(up_conv10): Sequential(
  (0): ConvTranspose2d(64, 32, kernel_size=(2, 2), stride=(2, 2))
  (1): ReLU(inplace=True)
(conv10): Sequential(
  (0): Conv2d(96, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
  (1): BatchNorm2d(32, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (2): ReLU(inplace=True)
(conv11): Conv2d(32, 2, kernel_size=(1, 1), stride=(1, 1))

```

3.1.3 Μοντέλο Παραμετροποίησης και Μοντέλο Αξιολόγησης

Εμβαθύνοντας την ανάλυση της σημασιολογικής τμηματοποίησης προτού προχωρήσουμε στο επόμενο βήμα είναι να αναφέρουμε και να αναλύσουμε τα συστατικά και τους βασικούς κανόνες των νευρωνικών δικτύων και φυσικά να μελετήσουμε την τεχνική προκειμένου να προσδιορίσουμε την ακρίβεια του προτεινόμενου νευρωνικού δικτύου. Αρχικά, διερευνώνται οι δείκτες βελτιστοποίησης που χρησιμοποιούνται για τη μείωση του ποσοστού σφάλματος κατά την εκπαίδευση του μοντέλου UNet-VGG16. Κατά τη διάρκεια της εκπαίδευσης του νευρωνικού δικτύου, τα βάρη του αρχικοποιούνται τυχαία και στη συνέχεια ενημερώνονται σε κάθε επανάληψη εκπαίδευσης με τέτοιο τρόπο ώστε να αυξάνουν τη συνολική ακρίβεια του δικτύου.

Σε κάθε επανάληψη εκπαίδευσης, η έξοδος των δεδομένων εκπαίδευσης συγκρίνεται με τα πραγματικά δεδομένα με τη βοήθεια της συνάρτησης απώλειας για τον υπολογισμό του σφάλματος και στη συνέχεια το βάρος ενημερώνεται ανάλογα. Αυτό το πρόβλημα μέτρησης απόδοσης έχει ως στόχο της βελτιστοποίηση της λειτουργία απώλειας για να λαμβάνει τα ιδανικά βάρη. Η μέθοδος που χρησιμοποιείται για τη βελτιστοποίηση ονομάζεται optimizer. Για το συγκεκριμένο μοντέλο μας ο δείκτης βελτιστοποίησης που χρησιμοποιείται ονομάζεται Adam [21]. Ο δείκτης Adam Optimizer χρησιμοποιεί τόσο την ορμητική πρακτική όσο και τον προσαρμοστικό ρυθμό μάθησης για καλύτερη σύγκλιση ως προς την επιθυμητή έξοδο.

Όπως προαναφέρθηκε η ανάλυση της σημασιολογικής τμηματοποίησης είναι βασική διαδικασία πίσω από αυτούς τους αλγόριθμους ώστε να καθοριστεί εικονοστοιχείο προς εικονοστοιχείο με τη σωστή κλάση και μετά τη σύγκρισή τους. Τα μοντέλα μηχανικής μάθησης χρειάζονται έναν τρόπο εκτίμησης της ακρίβειας αυτών των αναγνωριστικών εικονοστοιχείων ανά εικονοστοιχεία. Το πρόβλημά μας είναι ο αλγόριθμος ταξινόμησης και η έξοδος του μοντέλου να είναι δύο κανάλια εξόδου. Κάθε κανάλι περιγράφει μία κλάση, στο παράδειγμά μας, έχουμε ένα σύνολο δύο κλάσεων.

Ο πρώτος είναι ο παίκτης και ο δεύτερος είναι το φόντο. Όταν έχουμε δύο πιθανές εξόδους το πρόβλημα είναι δυαδικό και υπάρχουν δύο τρόποι για να βρούμε την απώλεια. Η μέθοδος είναι η εξίσωση που ονομάζεται διασταυρούμενη εντροπία. Τουναντίον, στην περίπτωση της μία εξόδου η απώλεια που θα χρησιμοποιηθεί ονομάζεται δυαδική διασταυρούμενη εντροπία. Η απώλεια διασταυρούμενης εντροπίας χρησιμοποιείται κατά την προσαρμογή των βαρών του μοντέλου κατά τη διάρκεια της εκπαίδευσης. Ο στόχος είναι να ελαχιστοποιηθεί η απώλεια, πιο συγκεκριμένα όσο μικρότερη είναι η απώλεια τόσο καλύτερο είναι το μοντέλο. Ένα τέλειο μοντέλο έχει απώλεια διασταυρούμενης εντροπίας μηδέν. Η διασταυρούμενη εντροπία ορίζεται στην εικόνα 13:

$$L_{CE} = - \sum_{i=1}^n t_i \log(p_i), \text{ for } n \text{ classes,}$$

where t_i is the truth label and p_i is the Softmax probability for the i^{th} class.

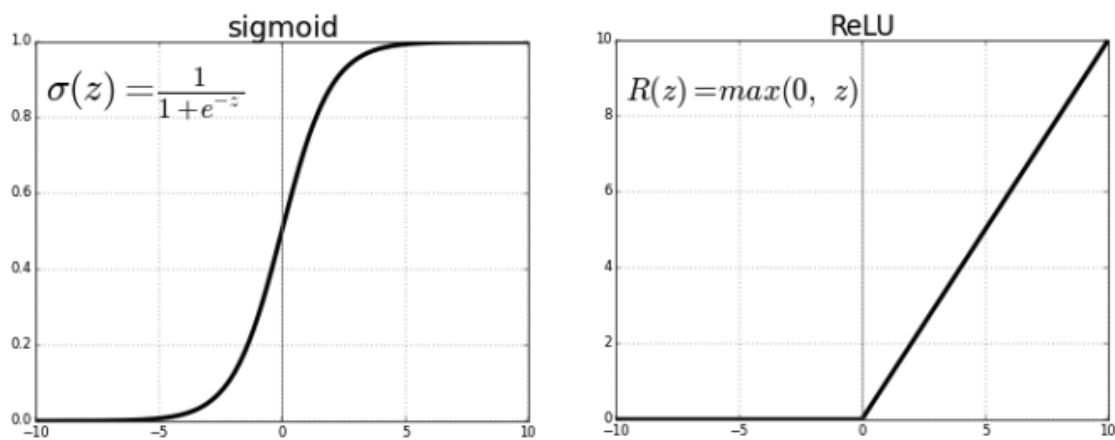
Εικόνα 13: Μαθηματικός ορισμός της διασταυρούμενης εντροπίας. Σημειώστε ότι ο λογάριθμος υπολογίζεται στη βάση 2.

Τέλος τα χαρακτηριστικά που πρόκειται να αναλυθούν είναι οι συναρτήσεις ενεργοποίησης και η κανονικοποίηση μιας παρτίδας εικόνων εκπαίδευσης που χρησιμοποιείται επίσης στην τρέχουσα εργασία. Μια παρτίδα εικόνων εκπαίδευσης αποτελεί ουσιαστικό μέρος της βιβλιογραφίας υλοποίησης αλγορίθμων βαθιάς μηχανικής μάθησης [22]. Συχνά προστίθεται ως μέρος ενός συνελκτικού επιπέδου και βοηθά στη σταθεροποίηση του δικτύου κατά τη διάρκεια της εκπαίδευσης.

Πιο αναλυτικά είναι ένα πρόσθετο επίπεδο δικτύου που εισάγεται μεταξύ ενός κρυφού επιπέδου και του επόμενου κρυφού επιπέδου. Η δουλειά του είναι να πάρει τις εξόδους από το πρώτο κρυφό στρώμα και να τις ομαλοποιήσει πριν τις μεταβιβάσει ως είσοδο του επόμενου κρυφού στρώματος. Στην εφαρμογή μας έχουμε ορίσει την κανονικοποίηση παρτίδας στο στρώμα αποκωδικοποίησης μόνο καθώς ο

κωδικοποιητής έχει προκαθορισμένα βάρη. Η ανάλυση συνεχίστηκε στις συναρτήσεις ενεργοποίησης που έχουν χρησιμοποιηθεί.

Όπως είναι γνωστό από τις μέχρι τώρα μελέτες, στην υπολογιστική όραση και πιο συγκεκριμένα στις εικόνες υπάρχει μια λειτουργία μεγάλης προσαρμοστικότητας στα εικονοστοιχεία πεδίου, που ονομάζεται συνάρτηση ενεργοποίησης ReLU (Διορθωμένη Γραμμική Μονάδα) [23]. Γνωρίζοντας ότι τα εικονοστοιχεία έχουν εύρος τιμών μεταξύ $[0, 255]$, η συγκεκριμένη συνάρτηση μπορεί να μετατραπεί στην περιοχή $[0, \infty)$. Επίσης τις περισσότερες φορές στο τελικό στρώμα των νευρωνικών δικτύων οι συνδέσεις με τις εξόδους των λειτουργιών ενεργοποίησης που χρησιμοποιούνται είναι η σιγμοειδής συνάρτηση. Ο λόγος είναι ότι η σιγμοειδής συνάρτηση μετατρέπει τις εισερχόμενες αριθμητικές τιμές στο εύρος $[0,1]$. Στην εικόνα 14 παρουσιάζουμε τις διαφορές μεταξύ των χρησιμοποιούμενων λειτουργιών ενεργοποίησης:

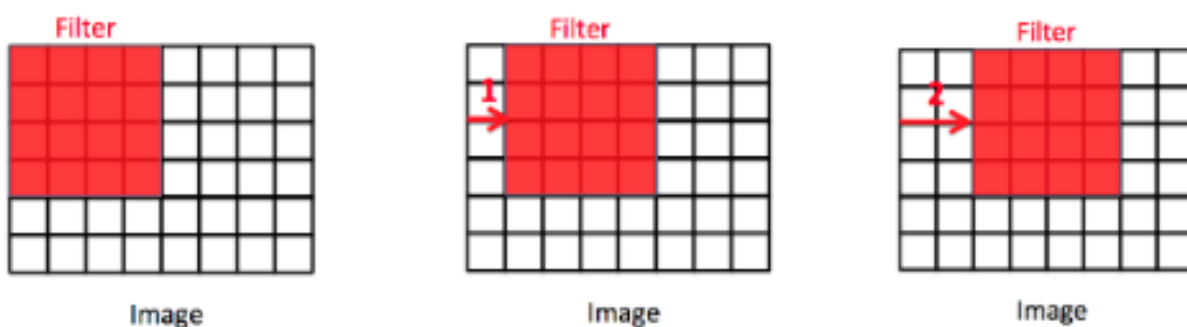


Εικόνα 14: ReLU v/s Σιγμοειδής

3.1.4 Αποτελέσματα Ακρίβειας Μοντέλου

Ο λόγος που δεν παρουσιάζουμε όλα τα αποτελέσματα των αλγορίθμων ως ενιαία οφείλεται στο γεγονός ότι οι αλγόριθμοι που υπολογίζουν την ακρίβεια κάθε προσέγγισης έχουν διαφορετική φιλοσοφία. Προτού εξάγουμε τα αποτελέσματα με την απόδοση αυτού του νευρωνικού δικτύου θα αναλύσουμε τις δυναμικές παραμέτρους αυτού του αλγορίθμου. Σε προηγούμενα μέρη, αναφέρθηκαν οι ημιστατικές παράμετροι που συνήθως είναι μη μεταβλητοί. Εν αντιθέση με την προαναφερθείσα αναγγελία θα αναφέρουμε παραμέτρους όπως ο διασκελισμός, η αναπλήρωση ορίων εικόνας, το μέγεθος του πυρήνα, οι επαναλήψεις εκπαίδευσης και ο ρυθμός εκπαίδευσης ενός νευρωνικού δικτύου.

Ο διασκελισμός είναι η διαδικασία που ο υπολογιστής διαβάσει μια εικόνα. Η ανάγνωση μιας εικόνας πραγματοποιείται από αριστερά προς τα δεξιά και από πάνω προς τα κάτω. Επομένως, ξεκινά από την επάνω αριστερή γωνία μέχρι την κάτω δεξιά γωνία. Επιπλέον, διαβάσει την εικόνα με συγκεκριμένο μέγεθος. Ο διασκελισμός ορίζει το βήμα που μετακινεί το φίλτρο από τη μία θέση στην επόμενη. Στο σενάριό μας της διαδικασία συνέλιξης στην οποία έχουμε θέσει διασκελισμό βήματος 1 και στην διαδικασία αναστροφής της συνέλιξης έχουμε ορίσει το βήμα διασκελισμού στο 2.



Εικόνα 15: Παράδειγμα Διασκελισμού

Η αναπλήρωση ορίων εικόνας είναι η προσθήκη επιπλέον εικονοστοιχείου εκτός από την εικόνα. Μηδενική αναπλήρωση σημαίνει ότι κάθε τιμή εικονοστοιχείου που προσθέτετε είναι μηδέν. Κάθε φορά που χρησιμοποιούμε το φίλτρο (γνωστό και ως πυρήνα) για να σαρώσουμε την εικόνα, το μέγεθος της εικόνας θα γίνεται όλο και μικρότερο. Επειδή δεν είναι επιθυμητό θέλουμε να διατηρήσουμε το αρχικό μέγεθος της εικόνας για να εξαγάγουμε ορισμένα χαρακτηριστικά χαμηλού επιπέδου. Επομένως, θα προσθέσουμε μερικά επιπλέον εικονοστοιχεία εκτός εικόνας. Σε αυτή την εργασία έχουμε προσθέσει επιπλέον εικονοστοιχεία μόνο στην οριζόντια συνέλιξη.

Image

0	0	0	0	0	0	0
0						0
0						0
0						0
0						0
0						0
0	0	0	0	0	0	0

Εικόνα 16: Η αναπλήρωση έχει οριστεί σε 1

Η συνέλιξη είναι ο πολλαπλασιασμός του πυρήνα (ή φίλτρου) και ενός τμήματος μιας εικόνας (τοπικό δεκτικό πεδίο) του ίδιου μεγέθους. Η συνέλιξη είναι αρκετά

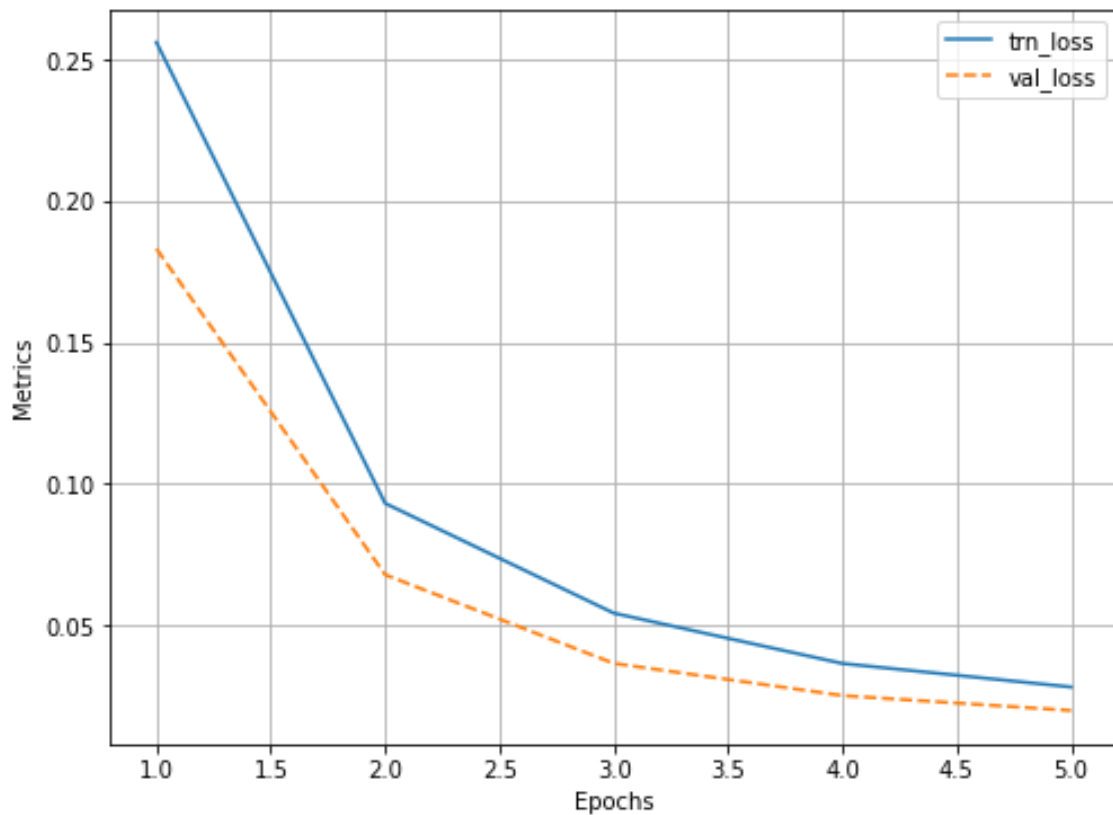
παρόμοια με τη συσχέτιση και παρουσιάζει μια ιδιότητα ισοδύναμης μετάφρασης που σημαίνει ότι αν μετακινήσουμε ή μεταφράσουμε την είσοδο και εφαρμόσουμε τη συνέλιξη σε αυτήν, θα ενεργήσει με τον ίδιο τρόπο που εφαρμόσαμε πρώτα τη συνέλιξη και στη συνέχεια μεταφράσαμε την εικόνα. Σε αυτή τη μελέτη οι διαστάσεις των πυρήνων που επιλέχθηκαν ήταν, για την οριζόντια συνέλιξη 3x3 και για την συνέλιξη αναστροφής ήταν 2x2.

Το ενδιαφέρον μέρος εδώ είναι η επιλογή της παραμέτρου των επαναλήψεων εκπαίδευσης και του ρυθμού εκπαίδευσης. Συνήθως όταν εκπαιδεύουμε βάρη σε συνελκτικά δίκτυα, χρειαζόμαστε χρόνο και φυσικά μεγαλύτερα βήματα. Προσπαθούμε να εξάγουμε όσο το δυνατόν λεπτομερέστερα περισσότερα μοτίβα μέσω των επαναλήψεων και του ρυθμού εκπαίδευσης. Οι επαναλήψεις είναι ο αριθμός που χρειάζεται ένα νευρωνικό δίκτυο για να κάνει μια πλήρη διάδοση πληροφορίας προς την έξοδο και ο ρυθμός οπισθοπορείας και εκπαίδευσης είναι το βήμα στο οποίο κάνουν οι δείκτες βελτιστοποίησης για να βρουν την κάθοδο χαμηλότερης κλίσης. Ο λόγος είναι ότι λόγω της μεταφοράς γνώσης και των προ-εκπαιδευμένων βαρών υπάρχει ανάγκη για εξαγωγή μοτίβων αφήνοντας τον αλγόριθμο να δημιουργήσει τη μάσκα ως αποκωδικοποιητή.

Από προηγούμενα παραδείγματα φτάνουμε στο συμπέρασμα ότι αυτές οι τιμές είναι αρκετά μικρές, ο ρυθμός εκπαίδευσης στο 0,001 και οι επαναλήψεις στο 5. Υπάρχουν διάφορες τεχνικές και δείκτες για τον έλεγχο της απόδοσης των νευρωνικών δικτύων. Χρησιμοποιήσαμε τον δείκτη με την ονομασία απώλεια με αποτελέσματα άκρως ικανοποιητικά για την αναγνώριση των παικτών.

EPOCH: 1.000	trn_loss: 0.256	trn_acc: 0.975	val_loss: 0.183	val_acc: 0.984	(689.48s - 2757.90s remaining)
EPOCH: 2.000	trn_loss: 0.093	trn_acc: 0.994	val_loss: 0.068	val_acc: 0.998	(1389.86s - 2084.80s remaining)
EPOCH: 3.000	trn_loss: 0.054	trn_acc: 0.994	val_loss: 0.036	val_acc: 0.998	(2088.19s - 1392.12s remaining)
EPOCH: 4.000	trn_loss: 0.036	trn_acc: 0.995	val_loss: 0.025	val_acc: 0.998	(2778.54s - 694.63s remaining)
EPOCH: 5.000	trn_loss: 0.028	trn_acc: 0.995	val_loss: 0.020	val_acc: 0.998	(3457.16s - 0.00s remaining)

Εικόνα 17: Απόλεια ανά επανάληψη



Εικόνα 18: Απόλεια επικύρωσης έναντι Απόλεια εκπαίδευσης

Οι εικόνες 17 και 18 αντιπροσωπεύουν την απόδοση του μοντέλου UNet-VGG16. Στην πρώτη εικόνα μπορούμε να παρατηρήσουμε την απόλεια εκπαίδευσης, την απόλεια επικύρωσης, την ακρίβεια της εκπαίδευσης, την ακρίβεια επικύρωσης και την περίοδο σε δευτερόλεπτα που απαιτείται για την εκπαίδευση του μοντέλου. Μπορούμε να εξαγάγουμε το ακόλουθο συμπέρασμα από τα τρέχοντα αποτελέσματα που έχουν

εκπληκτική απόδοση, οπότε αυτός είναι ένας παράγοντας επειδή δεν δοκιμάσαμε πολλαπλά πειράματα με άλλες παραμέτρους. Τέλος παρατηρούμε σε αυτό το μέρος αξίζει να παρατηρήσουμε από το τελευταίο σχήμα ότι για να μπορέσουμε να προσδιορίσουμε την αποδοτικότητα οφείλουμε να εξετάσουμε εάν το μοντέλο πρόκειται να υπερ εκπαιδευτεί ή να υπο εκπαιδευτεί. Από το διάγραμμα η τάση της απώλειας επικύρωσης μετά από μια χρονική περίοδο φαίνεται να είναι οριζόντια παράλληλη με τον άξονα x και αυτό σημαίνει ότι το μοντέλο δεν υπερ εκπαιδεύτηκε η δεν υπό εκπαιδεύτηκε.

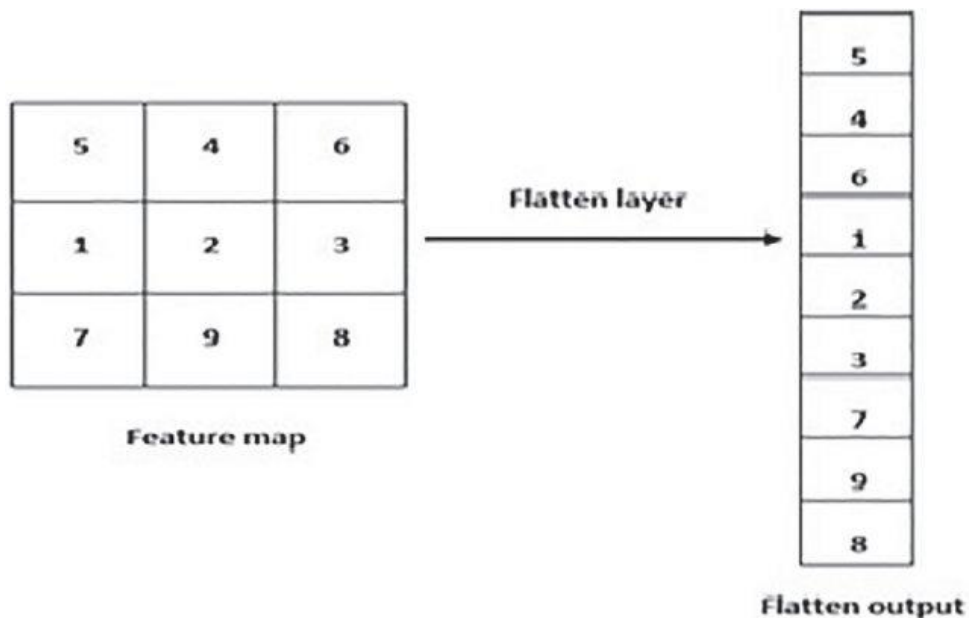
3.2 ΑΝΑΓΝΩΡΙΣΗ ΠΡΟΒΟΛΗΣ ΚΑΜΕΡΑΣ

Το επόμενο βήμα της προτεινόμενης λύσης είναι η ανίχνευση προβολής κάμερας. Αυτή η διαδικασία είναι πολύ σημαντική γιατί λαμβάνεται κρίσιμη απόφαση είτε θα προχωρήσουμε με την αναγνώριση του σχηματισμού είτε η προβολή της κάμερας δεν ικανοποιεί τις απαιτήσεις για έγκυρα συμπεράσματα.

3.2.1 Σύνολο Δεδομένων και Προεπεξεργασία Δεδομένων

Η μεθοδολογία ούσα μια ακολουθία ενεργειών, το στάδιο το οποίο βρίσκεται η διαδικασία αποτελείται από μια ασπρόμαυρη εικόνα, αποκαλώντας την ως μάσκα που έχει αναπαραχθεί από το προηγούμενο βήμα. Το τρέχον πρόβλημα δεν έχει ως είσοδο σύνθετους χάρτες χαρακτηριστικών και δεν απαιτείται η χρήση συνελκτικών νευρωνικών δικτύων. Για αυτό το μοντέλο ο αριθμός των εικόνων που χρησιμοποιήθηκαν ως σύνολο δεδομένων για την κανονική προβολή είναι 553 και 944 για την πίσω όψη. Το δεδομένα διαιρέθηκαν σε αναλογία 70-30 ως δεδομένα εκπαίδευσης και δοκιμής. Η λύση του προβλήματος προϋποθέτει την διανυσματοποίηση των εικόνων σε μορφή που να μπορούν να εκπαιδευτούν ως μία παρτίδα μονής διάστασης.

Παρουσιάζοντας ένα παράδειγμα για να γίνει πιο κατανοητό το πως κωδικοποιήθηκαν οι εικόνες σε διάνυσμα παίρνουμε τη λέξη ρεκόρ. Η λέξη μπορεί να μετασχηματιστεί σε μία διάσταση και το αποτέλεσμα που αναγράφεται στα παρακάτω σχήματα απεικονίζει σειρές, επί της ουσίας συμπιέζουμε όλες τις τιμές σε μία και μόνο διάσταση. Αυτό πραγματοποιείται σε όλες οι εικόνες σε διανύσματα μιας διάστασης εμπεριέχοντας τα βασικά εικονοστοιχεία τους. Η διαδικασία των εικονοστοιχείων κάθε εικόνας που περιγράψαμε απεικονίζεται στην Εικόνα 19:



Εικόνα 19: Μετατροπή της δομής εικόνας σε επίπεδο ισοπέδωσης.

Μία ακόμη σημαντική παράμετρος που λάβαμε υπόψιν αφορούσε τον αριθμό των χαρακτηριστικών που θα περιέχει αυτός ο πίνακας. Ο αριθμός εξάγεται από την τρέχουσα μαθηματική εξίσωση:

Πλάτος x ύψος + 1. Όλες οι εικόνες έχουν τις ίδιες διαστάσεις 224 x 224 ως αποτέλεσμα του δικτύου VGG16 και αυτό το αποτέλεσμα πολλαπλασιασμού αυξήθηκε κατά ένα ως στήλη που αναρτάται η κλάση της εικόνας. Το επόμενο σχήμα αναπαριστά το αποτέλεσμα του πίνακα αυτής της διαδικασίας διανυσματοποίησης.

	0	1	2	3	4	5	6	7	8	9	...	50167	50168	50169	50170	50171	50172	50173	50174	50175	50176	
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0	1
1	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0	1
2	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0	1

Εικόνα 20: Πίνακας με διανυσματικές εικόνες.

3.2.2 Επιλογή Μοντέλου

Το συγκεκριμένο σενάριο αναγνώρισης προβολής κάμερας όπως προαναφέρθηκε ανήκει στην κατηγορία των δυαδικών προβλημάτων. Αυτό το αποτέλεσμα προέρχεται λόγω της θέσης που δίνει την δυνατότητα να έχουμε πανοραμική θέα που μας κατευθύνει στο επόμενο βήμα της απόφασης του σχηματισμού, στην αντίπερα όχθη το μοντέλο ανιχνεύει την μη πανοραμική θέα που σημαίνει ότι η στάση της κάμερας θα είναι στο πλάι ή πολύ κοντά ζουμ, ώστε να μην μπορούμε βγάλουμε συμπέρασμα αν ο σχηματισμός είναι ο κατάλληλος. Ο αλγόριθμος που χρησιμοποιείται σε αυτό το δυαδικό πρόβλημα είναι η λογιστική παλινδρόμηση. Αυτός ο τύπος στατιστικού μοντέλου (επίσης γνωστός ως μοντέλο logit) χρησιμοποιείται συχνά για ταξινόμηση και προγνωστική αναλυτική.

Εν συνεχεία, η λογιστική παλινδρόμηση εκτιμά την πιθανότητα να συμβεί ένα συμβάν, όπως για παράδειγμα ψηφίστηκε ή δεν ψηφίστηκε, με βάση ένα σύνολο δεδομένων ανεξάρτητων μεταβλητών. Δεδομένου ότι το αποτέλεσμα είναι μια πιθανότητα, η εξαρτημένη μεταβλητή οριοθετείται μεταξύ 0 και 1. Στην λογιστική παλινδρόμηση, εφαρμόζεται ένας μετασχηματισμός logit στις πιθανότητες — δηλαδή, η πιθανότητα επιτυχίας διαιρούμενη με την πιθανότητα αποτυχίας. Αυτό είναι επίσης κοινώς γνωστό ως log odds, ή ο φυσικός λογάριθμος των αποδόσεων, και αυτή η λογιστική συνάρτηση αντιπροσωπεύεται από τους ακόλουθους τύπους:

- **Logit(pi) = 1/(1+ exp(-pi))**
- **ln(pi/(1-pi)) = Beta_0 + Beta_1*X_1 + ... + B_k*K_k**

Σε αυτή την εξίσωση λογιστικής παλινδρόμησης, η μεταβλητή $\text{logit}(\pi_i)$ είναι η εξαρτημένη ή μεταβλητή απόκρισης και x είναι η ανεξάρτητη μεταβλητή. Η παράμετρος βήτα, ή συντελεστής, σε αυτό το μοντέλο εκτιμάται συνήθως μέσω εκτίμησης μέγιστης πιθανότητας (MLE). Αυτή η μέθοδος ελέγχει διαφορετικές τιμές beta μέσω πολλαπλών επαναλήψεων για βελτιστοποίηση των αποδόσεων. Όλες αυτές οι επαναλήψεις παράγουν τη συνάρτηση $\log \text{likelihood}$ και η λογιστική παλινδρόμηση επιδιώκει να μεγιστοποιήσει την τιμή αυτής της συνάρτησης για να κάνει καλύτερη εκτίμηση παραμέτρων. Μόλις βρεθεί ο βέλτιστος συντελεστής (ή συντελεστές εάν υπάρχουν περισσότερες από μία ανεξάρτητες μεταβλητές), οι πιθανές πιθανοτικές λύσεις για κάθε παρατήρηση μπορούν να υπολογιστούν, να καταγραφούν και να αθροιστούν για να δώσουν μια προβλεπόμενη πιθανότητα κλάσης. Για την δυαδική ταξινόμηση, μια πιθανότητα μικρότερη από .5 θα προβλέψει την κλάση της μη πανοραμικής κάμερας ενώ μια πιθανότητα μεγαλύτερη από 0 θα λάβει την απόφαση ότι η κάμερα λήψης είναι πανοραμική.

Μετά τον υπολογισμό του μοντέλου, είναι οφείλουμε να ακολουθήσουμε την πρακτική να αξιολόγησης, πιο συγκεκριμένα πόσο καλά το μοντέλο προβλέπει την εξαρτημένη μεταβλητή. Η δοκιμασία Hosmer-Lemeshow είναι μια δημοφιλής μέθοδος για την αξιολόγηση της καταλληλότητας του μοντέλου.

3.2.3 Αποτελέσματα αξιολόγησης

Για την περαιτέρω ανάλυση ακρίβειας του μοντέλου, υπολογίζουμε τα αποτελέσματα που μας δίνει ο αλγόριθμος ταξινόμησης. Οι μετρήσεις που εξετάστηκαν

ήταν η ολοκληρωμένη ακρίβεια, η ακρίβεια προτεραιότητας, η ανάκληση και η βαθμολογία f1. Η μελέτη των παρακάτω διαγραμμάτων εξάγει το συμπέρασμα ότι λόγω της υψηλής ακρίβειας του μοντέλου δεν υπήρχε ανάγκη για περισσότερα πειράματα με διαφορετικούς αλγόριθμους

Ολοκληρωμένη Ακρίβεια

Η ολοκληρωμένη ακρίβεια είναι μια μέτρηση για την αξιολόγηση μοντέλων ταξινόμησης. Πιο αναλυτικά είναι το κλάσμα των προβλέψεων που το μοντέλο μας έλαβε σωστή απόφαση. Ο τύπος που περιγράφει αυτόν τον δείκτη απόδοσης περιγράφεται ως:

$$\text{Ακρίβεια} = \frac{\text{Αριθμός σωστών προβλέψεων}}{\text{Αριθμός συνολικών προβλέψεων}}$$

Για προβλήματα δυαδικής ταξινόμησης, η ακρίβεια μπορεί επίσης να υπολογιστεί με όρους θετικών και αρνητικών ως εξής:

$$\text{Ακρίβεια} = \frac{\text{TM} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

όπου TP, TN, FP, FN αντιστοιχούν σε αληθινά θετικά, αληθή αρνητικά, ψευδώς θετικά και ψευδώς αρνητικά αντίστοιχα.

Η ακρίβεια δεν είναι πάντα η σωστή μέτρηση αξιολόγησης, ειδικά όταν διαθέτουμε ένα σύνολο δεδομένων με ανισορροπία κλάσης, όπου υπάρχει σημαντική διαφορά μεταξύ του αριθμού των θετικών και αρνητικών ετικετών.

Μερική Ακρίβεια και ανάκληση

Η μερική ακρίβειας προσπαθεί να απαντήσει στο εξής ερώτημα: Ποιο ποσοστό των θετικών ταυτοποιήσεων ήταν στην πραγματικότητα σωστό;

Η ακρίβεια ορίζεται ως εξής:

$$\text{Ακρίβεια} = \frac{\text{TP}}{\text{TP} + \text{EP}}$$

Η ανάκληση επιδιώκει να απαντήσει στο ακόλουθο ερώτημα που είναι το ποσοστό των πραγματικών θετικών προβλέψεων που εντοπίστηκε σωστά.

Με μαθηματικούς όρους η ανάκληση ορίζεται ως εξής:

$$\text{Ανάκληση} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

F1-Σκορ

Στη στατιστική ανάλυση της δυαδικής ταξινόμησης, η βαθμολογία F ή μέτρηση F είναι ένα μέτρο της ακρίβειας ενός μίας δοκιμασίας που περνάει το μοντέλο. Υπολογίζεται από την ακρίβεια και την ανάκληση της δοκιμής αυτής. Η βαθμολογία F1 είναι ο αρμονικός μέσος όρος της ακρίβειας και της ανάκλησης:

$$F1 - \text{Βαθμολογία} = 2 * \frac{\text{ακρίβεια} * \text{ανάκληση}}{\text{ακρίβεια} + \text{ανάκληση}}$$

Τα αποτελέσματα της αναφοράς των πειραμάτων ταξινόμησης παρουσιάζονται στον παρακάτω πίνακα:

	precision	recall	f1-score	support
0	0.99	0.99	0.99	280
1	0.98	0.98	0.98	169
accuracy			0.98	449
macro avg	0.98	0.98	0.98	449
weighted avg	0.98	0.98	0.98	449

Εικόνα 21: Έκθεση ταξινόμησης.

Η αποτελεσματικότητα του μοντέλου μας οδηγεί στο συμπέρασμα ότι η απόφαση σχετικά με την επόμενη ενέργεια από την ανίχνευση στάσης της κάμερας είναι ακριβής. Πράγμα που σημαίνει ότι μετά τα δύο προηγούμενα μοντέλα που προηγήθηκαν μπορούμε να προχωρήσουμε με τον υπολογισμό της τελικής δράσης που είναι η αναγνώριση του σχηματισμού.

3.3 ΑΝΑΓΝΩΡΙΣΗ ΣΧΗΜΑΤΙΣΜΟΥ 4-4-2

Το τρίτο και τελευταίο μέρος της εργασίας αφορά το πρόβλημα που προσπαθούμε να επιλύσουμε το οποίο είναι, εάν μία ομάδα βρίσκεται σε καθεστώς σχηματισμού 4-4-2. Είναι εύλογο το ερώτημα πως θα φτάσουμε σε αυτήν την απόφαση και αυτό έρχεται να μας λύσει μια ακόμη δυαδική ταξινόμηση λαμβάνοντας ως την μια κλάση τον επιθυμητό σχηματισμό και ως δεύτερη κλάση το αντίθετο από αυτόν τον σχηματισμό. Ακόμη για να αμβλύνουμε το πρόβλημα της αναγνώρισης των παικτών αποφασίσαμε να εφαρμόσουμε αυτό το στάδιο σε διαφορετική ακολουθία εικόνων αποσυνθέτοντας τη βιντεοσκόπηση διαφορετικής ομάδας αλλά παρόμοιας απόχρωσης εμφανίσεων.

3.3.1 Σύνολο Δεδομένων και Προεπεξεργασία

Πιο αναλυτικά λήφθηκε υπόψιν μία βιντεοσκόπηση από την οποία κάναμε εξαγωγή 1,101 εικόνες για τον σχηματισμό 4-4-2 και 1,722 εικόνες με διαφορετικούς σχηματισμούς. Επιπλέον η μέθοδος που χρησιμοποιήθηκε σε αυτό το στάδιο για την διανυσματοποίηση των μασκών είναι πανομοιότυπος με το προηγούμενο βήμα δηλαδή μετατροπή των διαστάσεων της εικόνας σε μία διάσταση. Ένα παράδειγμα ένα νέο στιγμιότυπο με μία μάσκα από το νέο σύνολο δεδομένων απεικονίζουν οι παρακάτω εικόνες.



Εικόνα 22: Μάσκα συνόλου δεδομένων 3^{ου} αλγορίθμου.

3.3.2 Επιλογή Μοντέλου Τρίτου Σταδίου και Αποτελέσματα

Όπως καταλαβαίνουμε, εφόσον πρόκειται για παρόμοιο πρόβλημα με το προηγούμενο στάδιο λόγω ότι έχουμε δυαδική ταξινόμηση δεν θα επαναληφθεί περαιτέρω ανάλυση καθώς το μόνο που θα μεταβάλλουμε είναι το σύνολο δεδομένων που εκπαιδεύουμε το μοντέλο. Ο αλγόριθμος που επιλέξαμε είναι ομοίως η λογιστική παλινδρόμηση. Επομένως, θα παρουσιάσουμε τα αποτελέσματα της απόδοσης του μοντέλου βάση της αναφοράς ταξινόμησης.

	precision	recall	f1-score	support
0	0.88	0.90	0.89	532
1	0.82	0.78	0.80	315
accuracy			0.86	847
macro avg	0.85	0.84	0.85	847
weighted avg	0.86	0.86	0.86	847

Εικόνα 23: Αναφορά απόδοσης μοντέλου.

Χρησιμοποιώντας τις μετρήσεις που επεξηγήθηκαν στην προηγούμενη ενότητα παρατηρούμε την μεγάλη ακρίβεια που λαμβάνουμε από το συγκεκριμένο μοντέλο που αποτελεί βάση για την ανάρτηση ως μία εργασία με πάρα πολύ επιτυχημένα ποσοστά ακρίβειας για βίντεο αγώνων με ομάδες που φοράνε εμφανίσεις κόκκινων χρωματισμών.

ΚΕΦΑΛΑΙΟ 4

ΠΕΙΡΑΜΑΤΑ ΚΑΙ ΠΑΡΟΥΣΙΑΣΗ

Η συγκεκριμένη έρευνα έχει πολύ ενδιαφέρον καθώς τα πολλαπλά κομμάτια που τη συνθέτουν κρύβει από πίσω μηχανική μάθηση και αξίζει να παρουσιάσουμε κάποια στιγμιότυπα από τον κάθε αλγόριθμο μεμονωμένα αλλά και συνδυαστικά.

4.1 ΕΠΙΔΕΙΞΗ ΑΝΙΧΝΕΥΣΗΣ ΠΑΙΚΤΗ

Για να οπτικοποιήσουμε και να εφαρμόσουμε την αρχιτεκτονική που προτείναμε χρησιμοποιήσαμε μερικά δείγματα ολιγόλεπτων βίντεο κατά αποτελέσματα είναι ορατά παρακάτω. Στην εικόνα 24 παρατηρούμε ότι ακόμα και σε μια πανοραμική μακρινή λήψη το μοντέλο μας είναι σε θέση να εξάγει την μάσκα.



Εικόνα 24: Ολόκληρος ο σχηματισμός

Στην επόμενη εικόνα βλέπουμε ότι λόγω της λήψης και της κάμερας δεν θα ήμασταν σε θέση να αναγνωρίσουμε τον σχηματισμό. Η μάσκα παρόλα αυτά φαίνεται να είναι ακριβής καθώς τα εικονοστοιχεία είναι σωστά καταναμημένα στο σχήμα των παικτών μας με την πορτοκαλί εμφάνιση. Δύο παίκτες βρίσκονται στην προβολή κάμερας και η μάσκα τους αναπαριστά λεπτομερώς.



Εικόνα 25: Διαφορετική προβολή μάσκας

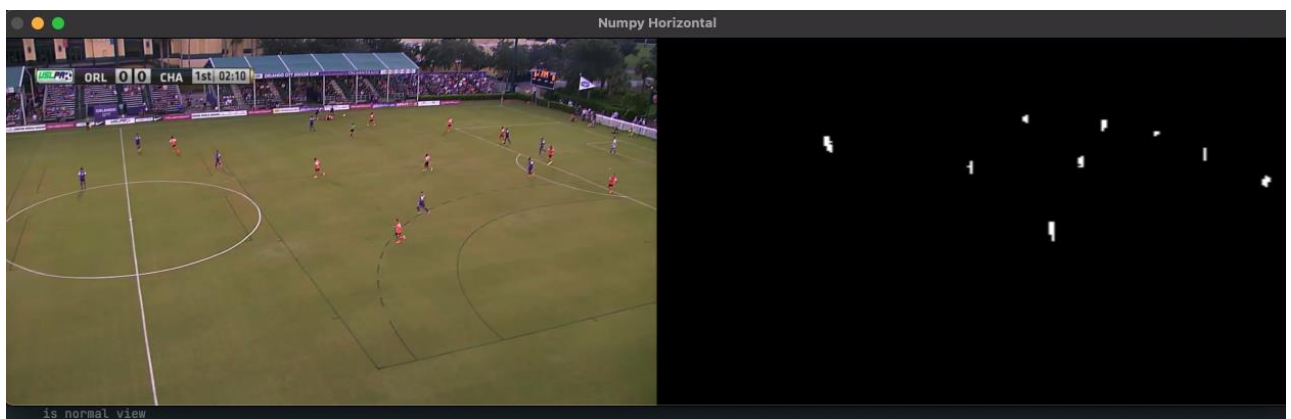
Στην εικόνα 26 έχουμε ένα ενδιαφέρον συμπέρασμα καθότι βλέπουμε ότι ο τερματοφύλακας δεν επηρεάζει την εξαγωγή της μάσκας μας. Με αυτόν τον τρόπο διαχωρίσαμε επίσης και τους τερματοφύλακες διότι δεν επηρεάζουν τον σχηματισμό μιας ομάδας εάν είναι ο σωστός. Επίσης αξίζει να σημειώσουμε ότι λόγω των πορτοκαλί εικονοστοιχείων σε ορισμένα μέρη της εικόνας υπάρχουν λευκά αποτυπώματα στη μάσκα. Αυτές οι μάσκες θα εξαιρεθούν από το επόμενο στάδιο της διαδικασίας καθότι δεν θα περάσουν τον στόχο της κανονικής λήψης.



Εικόνα 26: Μάσκα τερματοφύλακα

4.2 ΑΝΑΓΝΩΡΙΣΗ ΠΡΟΒΟΛΗΣ ΚΑΜΕΡΑΣ

Σε αυτό το σημείο παρουσιάζουμε την ακρίβειας του μοντέλου της δυαδικής ταξινόμησης σχετικά με την ανίχνευση προβολής κάμερας. Οι επερχόμενες εικόνες παρουσιάζουν τα αποτελέσματα για να αποδείξουν την ακριβή απόδοση αυτού του σημείου. Επιπλέον, όπως αναφέραμε και στο κεφάλαιο όπου αναλύσαμε λεπτομερώς την λήψη της κάμερας υπάρχουν δύο ενέργειες μετά από αυτό το βήμα, η πρώτη εάν η κάμερα είναι πανοραμική η μάσκα είναι έτοιμη να προχωρήσει στην αναγνώριση σχηματισμού και η δεύτερη τερματίζει τη διαδικασία ανάλυσης και προχωρά στο επόμενο πρόβλημα.

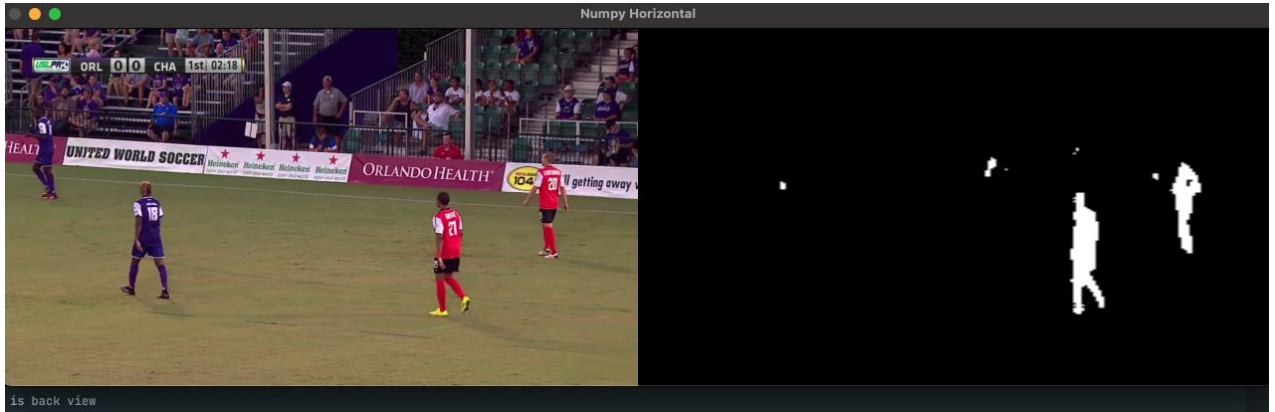


Εικόνα 27: Δείγμα πανοραμικής προβολής 1



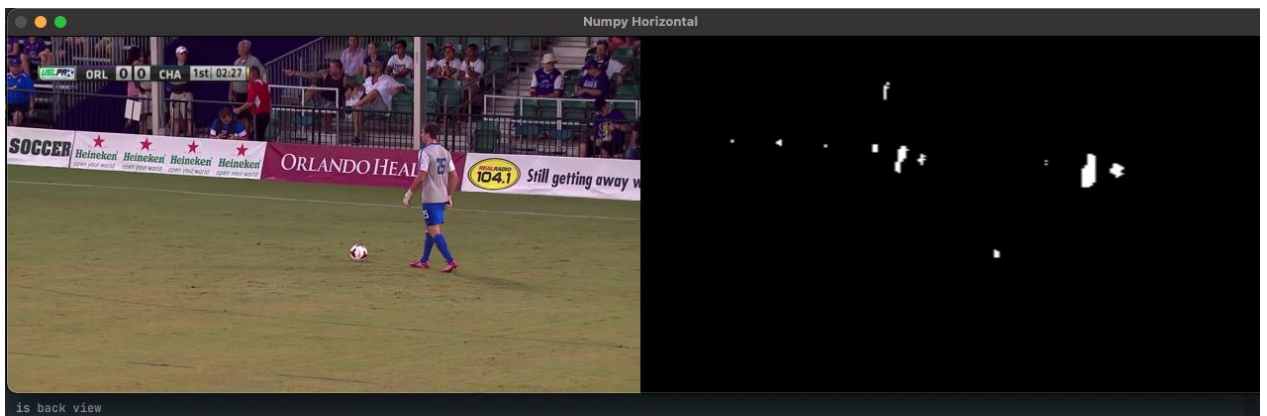
Εικόνα 28: Δείγμα πανοραμικής προβολής 2

Στις επόμενες δύο εικόνες λαμβάνουμε απόφαση εγκατάλειψης της διαδικασίας ανάλυσης.



Εικόνα 29: Δείγμα πίσω όψης 1

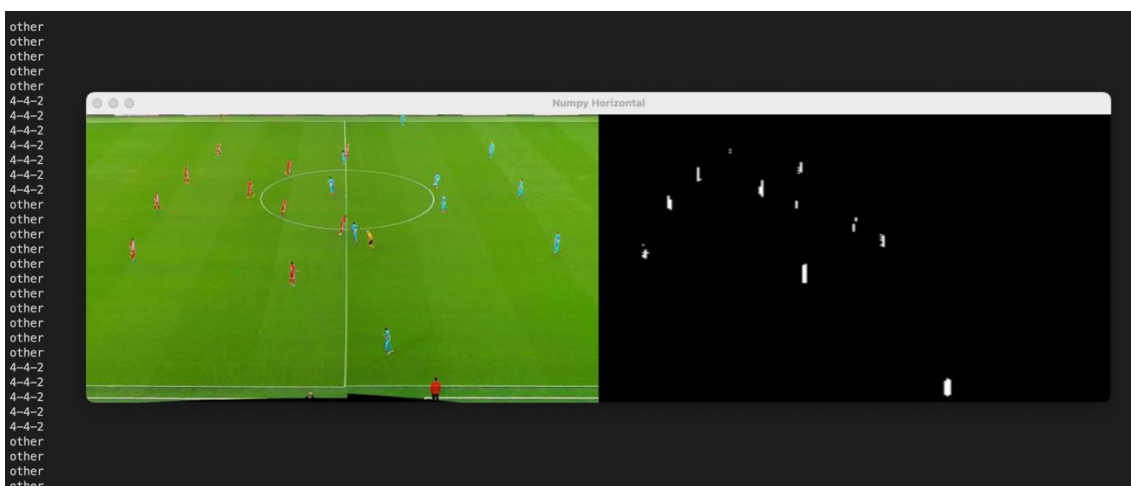
Βλέπουμε άλλο ένα παράδειγμα στην εικόνα 30 όπου εάν δεν είχαμε το φιλτράρισμα από το μοντέλο ανίχνευσης κάμερας θα μας οδηγούσε σε λάθος συμπεράσματα διότι παρατηρούμε ότι η μάσκα περιέχει αρκετά λευκά εικονοστοιχεία και το τρίτο στάδιο θα περιείχε μεγάλη αποτυχία.



Εικόνα 30: Δείγμα πίσω όψης 2

4.3 ΑΝΑΓΝΩΡΙΣΗ ΣΧΗΜΑΤΙΣΜΟΥ 4-4-2

Ο στόχος της αναγνώρισης σχηματισμού παρουσιάζεται με εικόνες από τα πειράματα που καταγράψαμε. Ο αναλυτής μπορεί σε αυτό το επίπεδο να κάνει εκτίμηση με τη βοήθεια του μοντέλου εάν ο σχηματισμός συμβαδίζει με την διάταξη τεσσάρων αμυντικών, τεσσάρων μέσων και δύο επιθετικών. Για να φτάσουμε σε αυτό το σημείο προϋποθέτει η κάμερα να είναι σε θέση να βλέπει όλους τους ποδοσφαιριστές .



Εικόνα 31: Σχηματισμός διαφορετικός

Στην εικόνα 31 οφείλουμε να την σχολιάσουμε και να εστιάσουμε σε μερικά σημαντικά σημεία. Καταρχήν βλέπουμε ο σχηματισμός στην προκείμενη εικόνα φέρνει τον σχηματισμό 4-4-2 παρά ταύτα το μοντέλο εμφανίζει συμπέρασμα ότι η διάταξη των παικτών είναι διαφορετική. Αυτό οφείλεται στη μάσκα που όπως παρατηρούμε ανιχνεύει τον διαιτητή λόγω της αποχρώσεως της φανέλας. Το μοντέλο λήψης σχηματισμού με την σειρά του φαίνεται ότι αποδίδει καλά μιας και η διάταξη είναι 4-4-2 εάν χρησιμοποιήσουμε τον διαιτητή στη μάσκα μας αλλά το πρόβλημα στο συγκεκριμένο πείραμα αφορά το μοντέλο εντόπισης των παικτών μας και όχι του σχηματισμού.



Εικόνα 32: Διάταξη 4-4-2

Τέλος, στην εικόνα 32 έχουμε επικύρωση της ακρίβειας του μοντέλου προσθέτοντας και προκατειλημμένη πληροφορία. Πιο αναλυτικά το αποτέλεσμα του μοντέλου απεικονίζει διάταξη 4-4-2 και ορθώς εάν παρατηρήσουμε την εικόνα έχουμε δύο γραμμές της άμυνας και του κέντρου και προωθημένους δύο ποδοσφαιριστές. Όπως αναφέραμε έχουμε και έκχυση κακής πληροφορία μιας και η μάσκα εντοπίζει και τον προπονητή της ομάδας στα όρια της φωτογραφίας. Παρόλα αυτά δεν λαμβάνει υπόψιν αυτή την παράμετρο της μάσκας.

ΚΕΦΑΛΑΙΟ 5

ΣΥΖΗΤΗΣΗ

Σε αυτήν τη μεταπτυχιακή εργασία, εξετάσαμε τα βήματα αναγνώρισης σχηματισμού ποδοσφαίρου που εφάπτεται τον τομέα της αθλητικής ανάλυσης. Αυτή η ολοκληρωμένη αρχιτεκτονική ακολουθιών λήψης αποφάσεων αποτελεί πρωτότυπη εργασία καθώς δεν υπάρχει βιβλιογραφικά άλλη όμοια μεθοδολογία, καθώς οι περισσότερες τεχνικές δίνουν προσοχή σε κάθε στοιχείο ξεχωριστά.

Η ανίχνευση αντικειμένων είναι ένα κοινό και συνηθισμένο πρόβλημα στην υπολογιστική όραση και πολλά ερευνητικοί τομείς εντείνουν την προσοχή τους σε αυτά τα θέματα καθότι μπορούν να εφαρμοστούν στον πραγματικό κόσμο και να δώσουν αξία στον επιχειρηματικό τομέα. Αυτό που επιχειρήσαμε ήταν μια ολοκληρωμένη λύση, ώστε να μπορεί να χρησιμοποιηθεί όχι μόνο σε άλλες ερευνητικές μελέτες αλλά και από αναλυτές και ειδικούς στο ποδόσφαιρο. Επίσης, αυτή η εργασία παρουσίασε μια πρακτική που είναι σε θέση να αναγνωρίζει όχι μόνο αντικείμενα αλλά και συγκεκριμένα αντικείμενα. Βέβαια αυτό απαιτεί πολύ χρόνο πριν από την επεξεργασία δεδομένων ως εισόδους σε αλγόριθμους. Τοπική προσαρμογή παίκτη είναι ένα πολύ σημαντικό αποτέλεσμα της σημασιολογικής τμηματοποίησης.

Η αναγνώριση λήψης κάμερας αποτελούσε το δεύτερο μέρος και υπάρχουν μερικά ενδιαφέροντα σημεία που μπορούν να συζητηθούν. Πρώτα απ' όλα, κατανοούμε τη σημασία αυτού του θέματος από την πλευρά του αναλυτή. Φανταστείτε αν παρακολουθείτε έναν αγώνα ποδοσφαίρου μέσα από το γήπεδο και η θέση σας είναι σε κακή θέση, το οποίο έχει ως αποτέλεσμα να μην μπορείτε να παρακολουθήσετε πολύ καλά τον αγώνα, γι' αυτό οι θέσεις στη μέση είναι πιο ακριβές επειδή επηρεάζουν την ποιότητα της παρακολούθησής σας. Δεύτερον, σε πολλές ερευνητικές μελέτες έχουμε δει ότι υπάρχουν τεχνικές βαθμονόμησης κάμερας και αναδημιουργίας λήψεων τριών διαστάσεων και προβολών σε δυοδιάστατα δείγματα. Στη συγκεκριμένη εργασία δεν

μας επηρεάζει αυτό το πρόβλημα επειδή η πανοραμική προβολή είναι σταθερή, οπότε ο αλγόριθμος δεν χρειάζεται να ταιριάζει σε άλλες κάμερες σε πραγματικό χρόνο.

Τέλος, μπορούμε να συζητήσουμε ένα θέμα που είναι σημαντικό όχι από ερευνητική άποψη αλλά από επιχειρηματική και τεχνική άποψη. Όπως γνωρίζουμε η υπολογιστική όραση είναι ένα πεδίο που απαιτεί μεγάλους πόρους για την ανάλυση. Αυτό είναι και ορατό στην εφαρμογή μας καθώς παρακολουθούμε το δείγμα.

Υπάρχει καθυστέρηση στη διαδικασία που χειρίζεται όλα τα μοντέλα και τις ενέργειες. Αυτή η καθυστέρηση είναι αποτέλεσμα ότι η διαδικασία τρέχει σε ένα νήμα λόγω της ανάγνωσης των στιγμιότυπων βιντεοσκόπησης και φυσικά του ταυτοχρονισμού. Επιπλέον, εξετάζοντας τη μεθοδολογία προγραμματισμού καταλήγουμε στο συμπέρασμα ότι υπάρχουν πολλές υπο-ρουτίνες που έχουν δημιουργηθεί για να ικανοποιήσουν τη φιλοσοφία της ακολουθίας ενεργειών που ακολουθήσαμε.

ΚΕΦΑΛΑΙΟ 6

ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΕΣ ΕΡΓΑΣΙΕΣ

Στόχος της παρούσας διπλωματικής εργασίας, ήταν η μελέτη των αλγορίθμων μηχανικής μάθησης σημασιολογικής τμηματοποίησης και η χρήση μοντέλων μηχανικής μάθησης για τη δημιουργία μιας ολοκληρωμένης λύσης για αναλυτές και ειδικούς του ποδοσφαίρου.

Όπως προαναφέρθηκε, υπάρχουν ορισμένα θέματα που μπορούν να βελτιώσουν την απόδοση της ακολουθίας των ενεργειών που διενεργούνται. Πιο συγκεκριμένα, επειδή επικεντρωθήκαμε στην απόδοση των μοντέλων και όχι του κώδικα, είναι σημαντικό να αναφέρουμε κάποιες τεχνικές που μπορούν να χρησιμοποιηθούν ως μελλοντικές πρακτικές. Μία από αυτές τις τεχνικές είναι η χρήση παράλληλων διαδικασιών. Αυτό μπορεί να υλοποιηθεί με τη βοήθεια νημάτων και ουρών. Μπορούμε να στέλνουμε ως παραγωγοί δεδομένα σε ουρές και να καταναλώνουμε αυτά τα νήματα δεδομένων που ονομάζονται εργαζόμενοι παράλληλα, ως αποτέλεσμα αυτής της μεθοδολογίας η απόδοσης θα βελτιωθεί σημαντικά.

Επίσης συζητήθηκε η διάρκεια που χρειαστήκαμε για την προεπεξεργασία δεδομένων, σε μελλοντικές εργασίες θα είναι πολύ καλό να αφιερώσουμε περισσότερο χρόνο στη σχεδίαση πολυγώνων σε εικόνες και στη χρήση στιγμιότυπων από πολλαπλές κάμερες. Τέλος, μπορούμε να χρησιμοποιήσουμε εικόνες της ομάδας μας όχι μόνο από ένα συγκεκριμένο παιχνίδι αλλά από ποικιλία αγώνες.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] Linda G. Shapiro and George C. Stockman (2001): “Computer Vision”, pp 279–325, New Jersey, Prentice-Hall, ISBN 0-13-030796-3
- [2] Barghout, Lauren, and Lawrence W. Lee. "Perceptual information processing system." Paravue Inc. U.S. Patent Application 10/618,543, filed July 11, 2003. The Medical Futurist. The top 12 health chatbots, 2020.
- [3] Nielsen, Frank; Nock, Richard (2003). On region merging: The statistical soundness of fast sorting, with applications. IEEE. pp. II:19–26. doi:10.1109/CVPR.2003.1211447
- [4] Yi, Jingru; Wu, Pengxiang; Jiang, Menglin; Huang, Qiaoying; Hoepfner, Daniel J.; Metaxas, Dimitris N. (July 2019). "Attentive neural cell instance segmentation". *Medical Image Analysis*. 55: 228–240.
- [5] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, Piotr Dollár (2018). "Panoptic Segmentation".
- [6] Guo, Dazhou; Pei, Yanting; Zheng, Kang; Yu, Hongkai; Lu, Yuhang; Wang, Song (2020). "Degraded Image Semantic Segmentation With Dense-Gram Networks". *IEEE Transactions on Image Processing*. 29: 782–795.
- [7] Jacek Komorowski, Grzegorz Kurzejamski and Grzegorz Sarwas 2020 FootAndBall: Integrated player and ball detector
- [8] Lu, keyu and Chen, Jianhui and Little, James J. Light Cascaded Convolutional Neural Networks for Accurate Player Detection.
- [9] F Multimodal and multiview distillation for real-time player detection on a football field. Anthony Cioppa, Adrien Deliege, Noor Ul Huda
- [10] Using Convolutional Neural Networks in Robots with Limited Computational Resources: Detecting NAO Robots while Playing Soccer. Nicolás Cruz, Kenzo Lobos-Tsunekawa, and Javier Ruiz-del-Solar.
- [11] XNOR-Net: ImageNet Classification Using Binary Convolutional Neural Networks Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, Ali Farhadi

- [12] A SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size Forrest N. Iandola, Song Han, Matthew W. Moskewicz, Khalid Ashraf, William J. Dally, Kurt Keutzer
- [13] Soccer Jersey Number Recognition Using Convolutional Neural Networks *Sebastian Gerke, Karsten Muller, Ralf Schafer*. Proceedings of the IEEE International Conference on Computer Vision (ICCV) Workshops, 2015, pp. 17-24
- [14] A robust method for automatic player detection in sport videos A. Lehuger S. Duffner C. Garcia
- [15] Sports Camera Calibration via Synthetic Data Jianhui Chen and James J. Little
- [16] A Camera Calibration and Player Localization in SoccerNet-v2 and Investigation of their Representations for Action Spotting Anthony Cioppa Adrien Deliege Floriane Magera Silvio Giancola Olivier Barnich Bernard Ghanem Marc Van Droogenbroeck
- [17] An Automated System for Generating Tactical Performance Statistics for Individual Soccer Players from Videos Rajkumar Theagarajan, Student Member, IEEE, and Bir Bhanu, Life Fellow, IEEE
- [18] "Does 4-4-2 exist?" – An Analytics Approach to Understand and Classify Football Team Formations in Single Match Situations Eric Müller-Budack Theiner Robert Rein Ralph Ewerth
- [19] U-Net: Convolutional Networks for Biomedical Image Segmentation Olaf Ronneberger, Philipp Fischer, and Thomas Brox
- [20] C. Balakrishna, D. Sarshar, S. Soltaninejad, "Automatic detection of lumen and media in the IVUS images using U-Net with VGG16 Encoder," arXiv preprint arXiv:1806.07554, June 2018.
- [21] Adam: A Method for Stochastic Optimization Diederik P. Kingma, Jimmy Ba
- [22] Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift Sergey Ioffe Google Inc., com Christian Szegedy
- [23] A Deep Learning using Rectified Linear Units (ReLU) Abien Fred Agarap arXiv:1803.08375
- [24] [Panoptic Segmentation Alexander Kirillov^{1,2} Kaiming He¹ Ross Girshick¹ Carsten Rother² Piotr Dollar¹](#) ¹Facebook AI Research (FAIR) ²HCI/IWR, Heidelberg University, Germany