ΠΑΝΕΠΙΣΤΗΜΙΟ ΜΑΚΕΔΟΝΙΑΣ
ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΤΜΗΜΑΤΟΣ ΕΦΑΡΜΟΣΜΕΝΗΣ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΠΕΙΡΑΜΑΤΙΚΗ ΜΕΛΕΤΗ ΑΝΑΓΝΩΡΙΣΗΣ ΑΝΤΙΚΕΙΜΕΝΩΝ ΣΕ ΦΟΡΗΤΕΣ ΣΥΣΚΕΥΕΣ ΜΕ ΤΗ ΧΡΗΣΗ ΤΗΣ ΥΠΟΛΟΓΙΣΤΙΚΗΣ ΝΕΦΟΥΣ

Διπλωματική Εργασία

της
Σταμπολού Ανατολή

Θεσσαλονίκη, Νοέμβρης 2020

ΠΕΙΡΑΜΑΤΙΚΗ ΜΕΛΕΤΗ ΑΝΑΓΝΩΡΙΣΗΣ ΑΝΤΙΚΕΙΜΕΝΩΝ ΣΕ ΦΟΡΗΤΕΣ ΣΥΣΚΕΥΕΣ ΜΕ ΤΗ ΧΡΗΣΗ ΤΗΣ ΥΠΟΛΟΓΙΣΤΙΚΗΣ ΝΕΦΟΥΣ

Σταμπολού Ανατολή του Νικολάου
Πτυχίο Μηχανικού Βιομηχανικής Πληροφορικής ΤΕΙ Καβάλας, 2006

Διπλωματική Εργασία
υποβαλλόμενη για τη μερική εκπλήρωση των απαιτήσεων του

ΜΕΤΑΠΤΥΧΙΑΚΟΥ ΤΙΤΛΟΥ ΣΠΟΥΔΩΝ ΣΤΗΝ ΕΦΑΡΜΟΣΜΕΝΗ ΠΛΗΡΟΦΟΡΙΚΗ

Επιβλέπων Καθηγητής
Παπαδημητρίου Παναγιώτης

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 02/11/2020

Παναγιώτης Παπαδημητρίου, Ιωάννης Ρεφανίδης, Ελευθέριος Μαμάτας

Σταμπολού Ν. Ανατολή

Περίληψη

Η Υπολογιστική Νέφους (Cloud Computing)[1][2][3] και η Αναγνώριση Αντικειμένων (Objects Detection) [4][5] κατατάσσονται στις τεχνολογίες αιχμής. Η υπολογιστική νέφους έχει προσελκύσει το ενδιαφέρον καθώς δίνει λύσεις σε μεγάλες ανάγκες υπολογιστικής ισχύος που απαιτούνται από εφαρμογές του διαδικτύου των πραγμάτων (IoT) [6][7][8][9].

Στην παρούσα εργασία αναπτύσσεται μία Πειραματική Μελέτη Αναγνώρισης Αντικειμένων σε φορητές συσκευές με τη χρήση της Υπολογιστικής Νέφους. Το πείραμα για την αναγνώριση αντικειμένων χρησιμοποιεί ένα προ εκπαιδευμένο μοντέλο (pretrained model)[10][11], για την επανεκπαίδευση του μοντέλου (training model)[12][13] γίνεται σχολιασμός πεπερασμένων αριθμών νέων εικόνων (data set)[14] και το μοντέλο επανεκπαιδεύεται με την χρήση πλατφόρμας υψηλής κλίμακας μηχανικής μάθησης (Large-Scale Machine Learning)[15] [16] [17] [18][19][20] [21].

Η παραγωγική διαδικασία ώστε να καταγραφούν οι μετρήσεις απαιτεί τη διάθεση του μοντέλου (serving model) [22] [23] η οποία γίνεται σε έναν διακομιστή (edge computer) για την εξυπηρέτηση αιτημάτων πρόβλεψης. Τα αιτήματα πρόβλεψης αποστέλλονται στον διακομιστή επανειλημμένα για την πραγματοποίηση καταγραφών των δεικτών απόδοσης, χρόνος απόκρισης (response time), κατανάλωση επεξεργαστικής ισχύος (CPU) και μνήμης (memory). Τα πειραματικά μας αποτελέσματα δείχνουν την κατανάλωση πόρων του διακομιστή .

Λέξεις Κλειδιά: Υπολογιστική Νέφους, Εκπαίδευση Μοντέλου, Αναγνώριση Αντικειμένων, Υψηλής Κλίμακας Μηχανική μάθηση, Διάθεση Μοντέλου, Χρόνος Απόκρισης

Abstract

Cloud Computing and Object Detection are among the cutting-edge technologies. Cloud computing has attracted interest as it provides solutions to the great computing power needs of the Internet of Things (IoT) applications.

In the present work, an Experimental Study of Object Recognition in mobile devices using Cloud Computing is developed. The object recognition experiment uses a pre-trained model, the model is trained to comment on finite numbers of new images, and the model is retrained using a high-scale machine learning platform. (Large-Scale Machine Learning).

The production process to record the measurements requires the provision of the model (serving model) which is done on a server (edge computer) to serve forecast requests. Prediction requests are sent to the server repeatedly to record performance indicators, response time, CPU, and memory consumption. Our experimental results show the resource consumption of the server.

Keywords: Cloud Computing, Object Recognition, Model Training, High Scale Machine Learning, Model Mood, Response Time

Πρόλογος – Ευχαριστίες

Όπως είναι γνωστό η τεχνολογία και τα επιτεύγματα της είναι μια από της πλέον σημαντικές εξελίξεις που έχει να επιδείξει ο άνθρωπος στις αρχές αυτού του νέου αιώνα. Η έκρηξη της τεχνολογίας, έχει οδηγήσει αρκετούς τομείς της επιστήμης να απωλέσουν παραδοσιακές μεθόδους και εφαρμογές και να υιοθετήσουν νέες πρακτικές οι οποίες βασίζονται στην τεχνολογία.

Μία από τις πλέον σύγχρονες πρακτικές που έχουν αρχίσει να υιοθετούνται και να αναπτύσσονται είναι οι διαδικτυακές υπηρεσίες μέσω διαφόρων έξυπνων συσκευών. Μία από αυτές τις τεχνολογίες είναι και η αναγνώριση αντικειμένων, ανθρώπων και ζώων, σ' αυτή στηρίζεται και η παρούσα διατριβή.

Η επίτευξη της συγκεκριμένης διατριβής αποτέλεσε ένα νέο κεφάλαιο στο επιστημονικό μου υπόβαθρο τόσο σε θεωρητικό όσο και σε πρακτικό επίπεδο. Σε αυτό το σημείο θα ήθελα να ευχαριστήσω όλη την ομάδα του NetCloud του Πανεπιστημίου Μακεδονίας που εκτός από την υλικοτεχνική υποδομή για τα πειράματα, με την εμπειρία τους και με μοναδικό τρόπο ο καθένας συντέλεσαν στην περάτωση του πειράματος. Τον επιβλέποντα καθηγητή κο. Παπαδημητρίου Παναγιώτη για τη συνεχή καθοδήγηση και ενθάρρυνση, τα σχόλια και οι υποδείξεις του αποτέλεσαν πολύτιμα εφόδια. Πέρα όμως και πάνω από όλα, ευχαριστώ την οικογένειά μου, και κυρίως τον σύζυγό μου για την αμέριστη ψυχολογική υποστήριξη.

Περιεχόμενα

Πίνακας Εικόνων

Πίνακας Γραφημάτων

1 Εισαγωγή

Η εξέλιξη του διαδικτύου των πραγμάτων (IoT) και η εισχώρηση τους στην καθημερινότητα των χρηστών προκάλεσε σημαντικό ερευνητικό ενδιαφέρον για την υπολογιστική νέφους. Η υπολογιστική νέφους είναι ο υπολογισμός που λαμβάνει χώρα ή κοντά στη φυσική τοποθεσία είτε του χρήστη είτε της πηγής των δεδομένων. Τοποθετώντας τις υπηρεσίες υπολογιστών πιο κοντά σε αυτές τις τοποθεσίες, οι χρήστες επωφελούνται από ταχύτερες και πιο αξιόπιστες υπηρεσίες. Η υπολογιστική νέφους είναι ένας τρόπος ο οποίος μπορεί να χρησιμοποιηθεί και να διανεμηθεί μια κοινή ομάδα πόρων σε μεγάλο αριθμό τοποθεσιών. Στα πλαίσια αυτά το πεδίο εφαρμογής του και δεδομένου του συνεχώς αυξανόμενου αριθμού IoT σε φορητές συσκευές, η εκφόρτωση υπολογισμού αναδύεται ως αιχμή και σημαντική ερευνητική περιοχή με τεράστιες δυνατότητες και πρακτικές εφαρμογές, η υπολογιστική νέφους έχει προκαλέσει μεγάλο ερευνητικό ενδιαφέρον στην επιστήμη των υπολογιστών και στις τεχνολογίες αιχμής.

Άλλα πλεονεκτήματα της υπολογιστικής νέφους περιλαμβάνουν την ικανότητα διεξαγωγής μαζικών αναλυτικών στοιχείων και συγκεντρωτικών δεδομένων επί τόπου, κάτι που επιτρέπει τη λήψη αποφάσεων σχεδόν σε πραγματικό χρόνο. Μειώνει περαιτέρω τον κίνδυνο έκθεσης ευαίσθητων δεδομένων διατηρώντας όλη αυτή την υπολογιστική ισχύ τοπικά, επιτρέποντας έτσι στους παρόχους να ελέγχουν καλύτερα τη διάδοση των πληροφοριών (όπως τα εμπορικά μυστικά μίας βιομηχανίας) ή να τηρούν τις πολιτικές απορρήτου (όπως ο GDPR). Τέλος, οι χρήστες επωφελούνται από την ανθεκτικότητα και το κόστος που σχετίζεται.

Διατηρώντας την υπολογιστική ισχύ τοπικά, οι υπηρεσίες μπορούν να συνεχίσουν να λειτουργούν ανεξάρτητα από έναν κεντρικό υπολογιστή, ακόμη και αν κάτι προκαλεί την κεντρική υπηρεσία να σταματήσει να λειτουργεί. Το κόστος για τη μεταφορά των δεδομένων μεταξύ των πυρήνων και των περιφερειακών τοποθεσιών μειώνεται επίσης σημαντικά διατηρώντας αυτήν την υπολογιστική ισχύ πιο κοντά στην πηγή της.

Παράλληλα η ραγδαία ανάπτυξη του 5G το οποίο εξυπηρετεί εφαρμογές IoT προκαλεί και την ανάγκη για μεγαλύτερους υπολογιστικούς πόρους με αυξανόμενο ρυθμό ανάπτυξης της, με αποτέλεσμα να είναι μεγαλύτερη η απαίτηση αποσυμφόρησης του υπολογιστικού φορτίου.

Επίσης οι δυναμικές αλληλεπιδράσεις μεταξύ αυτών των εφαρμογών και των συσκευών είναι σημαντικός παράγοντας για την εξέλιξη της πολλαπλής πρόσβασης σε υπολογιστική νέφους (multiaccess edge computing (MEC)). Στην ουσία η συγκεκριμένη τεχνολογία αξιοποιεί την υπολογιστική ισχύ πολλαπλών συσκευών (όπως διακομιστές ή άλλους υπολογιστικούς πόρους κοντά σε σταθμούς βάσης, Wi-Fi ή σημεία πρόσβασης). Το MEC είναι ένα βασικό συστατικό για την πραγματοποίηση του Διαδικτύου των πραγμάτων (IoT) και των κινητών εφαρμογών σε έξυπνα περιβάλλοντα πόλης.

Τα παραπάνω αποτέλεσαν ερευνητικό πεδίο στο έργο COSMOS : An Orchestration Framework for Smart Computation Offloading in Edge Clouds. Το COSMOS αποτέλεσε μία αρχιτεκτονική σχεδίαση και μία πειραματική αξιολόγηση ενός πλαισίου ενορχήστρωσης για έξυπνη εκφόρτωση υπολογισμού από IoT κινητές συσκευές σε διακομιστές νέφους.

Επιπρόσθετα η χρήση υπολογιστικής νέφους αποτελεί πλέον βασική υποδομή λήψης και διαχείρισης δεδομένων που προέρχονται από υπηρεσίες μηχανικής μάθησης που εφαρμόζονται σε μία πληθώρα φορητών συσκευών όπως υπολογιστών, έξυπνων κινητών, έξυπνων ρολογιών, αυτοκινήτων κα. Μια έξυπνη πόλη χρησιμοποιεί διαφορετικές πηγές ηλεκτρονικών δεδομένων σε εφαρμογές Internet of things (IoT) για τη συλλογή δεδομένων. Οι γνώσεις των δεδομένα χρησιμοποιούνται για την αποτελεσματική διαχείριση πόρων και υπηρεσιών σε αντάλλαγμα, τα δεδομένα που χρησιμοποιούνται για να βελτιώνουν τις λειτουργίες σε όλη την πόλη. Αυτό περιλαμβάνει δεδομένα που συλλέγονται από πολίτες, συσκευές, κτίρια και περιουσιακά στοιχεία τα οποία στη συνέχεια υποβάλλονται σε επεξεργασία και αναλύονται για την παρακολούθηση και διαχείριση συστημάτων κυκλοφορίας και μεταφοράς, μονάδων παραγωγής ενέργειας, υπηρεσιών κοινής ωφέλειας, δικτύων ύδρευσης, απορριμμάτων, εντοπισμού εγκλημάτων, συστημάτων πληροφοριών, σχολείων, βιβλιοθηκών , νοσοκομεία και άλλες κοινοτικές υπηρεσίες.

Με βάση πολλές από τις παραπάνω υπηρεσίες αποστέλλουν δεδομένα που προέρχονται από ανίχνευση αντικειμένων μπορούν να εκμεταλλευτούν την υπολογιστική νέφους για την καλύτερη κατανομή και διαχείριση του υπολογιστικού φόρτου. Οι μέθοδοι για την ανίχνευση αντικειμένων εμπίπτουν γενικά σε προσεγγίσεις που βασίζονται σε μηχανική μάθηση Machine Learning (ML) ή σε προσεγγίσεις που βασίζονται σε βαθιά μάθηση Deep Learning (DL).

Η ανίχνευση αντικειμένων είναι μια τεχνολογία υπολογιστών που σχετίζεται το computer vision και την επεξεργασία εικόνας που ασχολείται με την ανίχνευση παρουσιών σημασιολογικών αντικειμένων μιας συγκεκριμένης κατηγορίας (όπως οι άνθρωποι, τα κτίρια ή τα αυτοκίνητα) σε ψηφιακές εικόνες και βίντεο. Οι τομείς που έχει γίνει μεγαλύτερη έρευνα και υπάρχει αποτελεσματικότητα ανίχνευσης αντικειμένων περιλαμβάνουν ανίχνευση προσώπου και ανίχνευση πεζών. Το computer vision και η δυνατότητα επεξεργασίας εικόνων, η ανίχνευση χαρακτηριστικών περιλαμβάνει μεθόδους για τον υπολογισμό των αφαιρετικών πληροφοριών εικόνας και τη λήψη τοπικών αποφάσεων σε κάθε σημείο εικόνας, εάν υπάρχει ένα χαρακτηριστικό εικόνας ενός δεδομένου τύπου σε αυτό το σημείο ή όχι. Τα χαρακτηριστικά που προκύπτουν θα είναι υποσύνολα του τομέα εικόνας, συχνά με τη μορφή απομονωμένων σημείων, συνεχών καμπυλών ή συνδεδεμένων περιοχών.

Κάθε κατηγορία αντικειμένων έχει τις δικές της ειδικές δυνατότητες που βοηθούν στην ταξινόμηση τους για παράδειγμα, όλοι οι κύκλοι είναι στρογγυλοί. Η ανίχνευση αντικειμένων χρησιμοποιεί αυτές τις ειδικές δυνατότητες. Για παράδειγμα, όταν αναζητάτε κύκλους,

αναζητούνται αντικείμενα που βρίσκονται σε μια συγκεκριμένη απόσταση από ένα σημείο (δηλαδή το κέντρο). Παρομοίως, όταν αναζητάτε τετράγωνα, απαιτούνται αντικείμενα που είναι κάθετα στις γωνίες και έχουν ίσα πλευρικά μήκη. Μια παρόμοια προσέγγιση χρησιμοποιείται για την αναγνώριση προσώπου όπου μπορούν να βρεθούν τα μάτια, η μύτη και τα χείλη και χαρακτηριστικά όπως το χρώμα του δέρματος και η απόσταση μεταξύ των ματιών.

1.1 Σύγχρονες Τεχνολογίες-Έξυπνη Ζωή

1.1.1 Εφαρμογές Διαδικτύου των Πραγμάτων (IoT)

1.1.1.1 Έξυπνο σπίτι

Οι συσκευές IoT αποτελούν μέρος της ευρύτερης έννοιας του αυτοματισμού στο σπίτι, που μπορεί να περιλαμβάνει φωτισμό, θέρμανση και κλιματισμό, μέσα ενημέρωσης και συστήματα ασφαλείας. Τα μακροπρόθεσμα οφέλη θα μπορούσαν να περιλαμβάνουν την εξοικονόμηση ενέργειας με την αυτόματη εξασφάλιση φώτων και τα ηλεκτρονικά συστήματα απενεργοποιούνται. Ένα έξυπνο σπίτι ή αυτοματοποιημένο σπίτι θα μπορούσε να βασίζεται σε μια πλατφόρμα ή κόμβους που ελέγχουν έξυπνες συσκευές και συσκευές. Για παράδειγμα, χρησιμοποιώντας το HomeKit της Apple, οι κατασκευαστές μπορούν να ελέγχουν τα οικιακά προϊόντα και τα εξαρτήματά τους από μια εφαρμογή σε συσκευές iOS, όπως το iPhone

1.1.1.2 Φροντίδα ηλικιωμένων

Μια βασική εφαρμογή ενός έξυπνου σπιτιού είναι η παροχή βοήθειας σε άτομα με αναπηρίες και ηλικιωμένους. Αυτά τα οικιακά συστήματα χρησιμοποιούν τεχνολογία υποβοήθησης για την αντιμετώπιση συγκεκριμένων αναγκών ενός ιδιοκτήτη. Ο φωνητικός έλεγχος μπορεί να βοηθήσει τους χρήστες με περιορισμούς όρασης και κινητικότητας, ενώ τα συστήματα συναγερμού μπορούν να συνδεθούν απευθείας με εμφυτεύματα που φοριούνται από χρήστες με προβλήματα ακοής. Μπορούν επίσης να εξοπλιστούν με πρόσθετα χαρακτηριστικά ασφαλείας. Αυτά τα χαρακτηριστικά μπορούν να περιλαμβάνουν αισθητήρες που παρακολουθούν για ιατρικές καταστάσεις έκτακτης ανάγκης όπως πτώσεις ή επιληπτικές κρίσεις. Η τεχνολογία Smart Home που εφαρμόζεται με αυτό τον τρόπο μπορεί να προσφέρει στους χρήστες περισσότερη ελευθερία και υψηλότερη ποιότητα ζωής.

1.1.1.3 Ιατρική και υγειονομική περίθαλψη

Το Διαδίκτυο των ιατρικών πραγμάτων (IoMT) είναι μια εφαρμογή του IoT για σκοπούς που σχετίζονται με την ιατρική και την υγεία, τη συλλογή και ανάλυση δεδομένων για έρευνα και παρακολούθηση. Το IoMT έχει αναφερθεί ως "έξυπνη υγειονομική περίθαλψη", ως τεχνολογία για τη δημιουργία ενός ψηφιακού συστήματος υγειονομικής περίθαλψης, συνδέοντας διαθέσιμους ιατρικούς πόρους και υπηρεσίες υγειονομικής περίθαλψης.

Οι συσκευές IoT μπορούν να χρησιμοποιηθούν για την ενεργοποίηση απομακρυσμένων συστημάτων παρακολούθησης της υγείας και ενημέρωσης έκτακτης ανάγκης. Αυτές οι συσκευές παρακολούθησης της υγείας μπορούν να κυμαίνονται από τις συσκευές παρακολούθησης της αρτηριακής πίεσης και του καρδιακού ρυθμού έως τις προηγμένες συσκευές που είναι σε θέση να παρακολουθούν εξειδικευμένα εμφυτεύματα, όπως βηματοδότες, ηλεκτρονικά βραχιολάκια Fitbit ή προηγμένα βοηθήματα ακοής.

Ορισμένα νοσοκομεία έχουν αρχίσει να εφαρμόζουν "έξυπνα κρεβάτια" που μπορούν να ανιχνεύσουν πότε είναι κατειλημμένα και όταν ένας ασθενής προσπαθεί να σηκωθεί. Μπορεί επίσης να προσαρμοστεί για να εξασφαλίσει την κατάλληλη πίεση και υποστήριξη στον ασθενή χωρίς τη χειρωνακτική αλληλεπίδραση των νοσοκόμων. Σύμφωνα με μια έκθεση του Goldman Sachs για το 2015, οι συσκευές IoT της υγειονομικής περίθαλψης «μπορούν να σώσουν τις Ηνωμένες Πολιτείες περισσότερα από 300 δισεκατομμύρια δολάρια σε ετήσιες δαπάνες υγειονομικής περίθαλψης, αυξάνοντας τα έσοδα και μειώνοντας το κόστος» Επιπλέον, η χρήση κινητών συσκευών για την υποστήριξη της ιατρικής παρακολούθησης οδήγησε η δημιουργία του «m-health», χρησιμοποίησε τις αναλυθείσες στατιστικές υγείας ». Οι εξειδικευμένοι αισθητήρες μπορούν επίσης να εξοπλιστούν σε χώρους διαβίωσης για την παρακολούθηση της υγείας και της γενικής ευημερίας των ηλικιωμένων, εξασφαλίζοντας ταυτόχρονα τη σωστή θεραπεία και βοηθώντας τους ανθρώπους να ανακτήσουν την απώλεια της κινητικότητας μέσω της θεραπείας. Αυτοί οι αισθητήρες δημιουργούν ένα δίκτυο έξυπνων αισθητήρων που είναι σε θέση να συλλέγουν, να επεξεργάζονται, να μεταφέρουν και να αναλύουν πολύτιμες πληροφορίες σε διαφορετικά περιβάλλοντα, όπως τη σύνδεση συσκευών παρακολούθησης στο σπίτι με συστήματα που βασίζονται σε νοσοκομεία. Άλλες καταναλωτικές συσκευές για την ενθάρρυνση της υγιούς διαβίωσης, όπως οι συνδεδεμένες ζυγαριές ή οι φορητές συσκευές παρακολούθησης της καρδιάς, είναι επίσης μια δυνατότητα με το IoT.

Οι πλατφόρμες IoT για την παρακολούθηση της υγείας είναι επίσης διαθέσιμες για τους προγεννητικούς και τους χρόνιους ασθενείς, βοηθώντας τους να διαχειριστούν τις υγιεινές ζωές και τις επαναλαμβανόμενες φαρμακευτικές απαιτήσεις. Οι πρόοδοι στις μεθόδους κατασκευής πλαστικών και ηλεκτρονικών υφασμάτων επέτρεψαν αισθητήρες IoMT εξαιρετικά χαμηλού κόστους, χρήσης και ρίψης. Αυτοί οι αισθητήρες, μαζί με τα απαιτούμενα ηλεκτρονικά

συστήματα RFID, μπορούν να κατασκευαστούν σε χαρτί ή σε ηλεκτρονικά προϊόντα για ασύρματες μονάδες ανίχνευσης μίας χρήσης. Έχουν τεθεί σε εφαρμογή εφαρμογές ιατρικής διάγνωσης, όπου η φορητότητα και η χαμηλή πολυπλοκότητα του συστήματος είναι απαραίτητες[26][28] [29].

1.1.1.4 Μεταφορές

Το IoT μπορεί να βοηθήσει στην ενοποίηση των επικοινωνιών, του ελέγχου και της επεξεργασίας πληροφοριών σε διάφορα συστήματα μεταφορών. Η εφαρμογή του IoT εκτείνεται σε όλες τις πτυχές των συστημάτων μεταφοράς (δηλαδή του οχήματος, της υποδομής και του οδηγού ή του χρήστη). Η δυναμική αλληλεπίδραση μεταξύ αυτών των στοιχείων ενός συστήματος μεταφορών επιτρέπει την ενδοκοινοτική και ενδοεπιχειρησιακή επικοινωνία, τον έξυπνο έλεγχο κυκλοφορίας, τον έξυπνο χώρο στάθμευσης, τα ηλεκτρονικά συστήματα είσπραξης διοδίων, την εφοδιαστική και τη διαχείριση του στόλου, τον έλεγχο των οχημάτων, την ασφάλεια και την οδική βοήθεια.

Καθώς αναπτύσσεται το Internet of Things, χρειαζόμαστε περισσότερο χώρο για να μπορούμε να επεξεργαστούμε τα πάντα. Έξυπνες πόλεις και αυτόνομα οχήματα έχουν μηδενική πιθανότητα να πραγματοποιηθούν εάν η επεξεργασία των δεδομένων πρέπει να γίνει σε επίπεδο cloud. Τα δεδομένα θα επεξεργάζονται σε πραγματικό χρόνο. Αυτό είναι κάτι που μόνο η τεχνολογία “Edge” μπορεί να υποστηρίξει αυτή τη στιγμή.

Το Edge Computing είναι έτοιμο να αλλάξει τη δικτύωση, όπως τη γνωρίζουμε σήμερα. Λόγω του Edge Computing, τα αυτόνομα οχήματα, τα έξυπνα σπίτια, ακόμη και οι έξυπνες πόλεις θα γίνουν τα νέα πρότυπα. Το cloud δεν είναι αρκετό και το Edge computing θα πρέπει να το αντικαταστήσει [30] [31].

1.1.2 Μηχανές Μάθησης Τεχνητής Νοημοσύνης

Κάθε αγορά και βιομηχανία φαίνεται να επενδύει μεγάλα χρήματα σε τεχνητή νοημοσύνη (AI) [35][36][37] και μηχανική μάθηση (ML)[17][18][33][34]. Σε cloud λογισμικό, διαδίκτυο των πραγμάτων (IoT), fintech, μεγάλα δεδομένα (Big Data) [38][39], ανεξάρτητα από το που βλέπετε, η τεχνητή νοημοσύνη είναι εκεί . Στις περισσότερες περιπτώσεις, όμως, έχουμε μόνο αγγίξει την επιφάνεια του τι μπορεί να κάνει η τεχνητή νοημοσύνη AI για εμάς. Χρησιμοποιούμε μόνο τις πιο βασικές λειτουργίες, δηλαδή την αυτοματοποίηση, η οποία εκτελεί εργασίες γρηγορότερα, με μεγαλύτερη ακρίβεια και με πολύ χαμηλότερο κόστος από κάθε άνθρωπο. Ένα από τα θεμελιώδη χαρακτηριστικά του 5G είναι η δυνατότητα πρόβλεψης της

δραστηριότητας στο δίκτυο και η απρόσκοπτη διαχείριση του. Η μηχανική μάθηση είναι ιδανική για να εργάζεται σε δίκτυα 5G, επειδή απαιτεί τεράστιες ποσότητες δεδομένων για την πρόβλεψη της δραστηριότητας με ακρίβεια.

Οι μηχανές μάθησης (Learning Machines) με εφαρμογές σε πεδία της καθημερινότητας όπως: Αναγνώριση Ομιλίας, Αναγνώριση Γραφικού Χαρακτήρα, Διαδικτυακή Διαφήμιση, Εντοπισμός Απάτης Πιστωτικής Κάρτας, Μηχανές Αναζήτησης, Αναγνώριση Εικόνων και αντικειμένων. Σε αυτό το πλαίσιο παρουσιάζεται η ικανότητα μιας μηχανής μάθησης να αποδίδει με ακρίβεια σε καινούριες, πρωτόγνωρες εργασίες, αφού πρώτα έχει εκπαιδευτεί σε ένα σύνολο δεδομένων εκπαίδευσης. Γενικά τα προς εκπαίδευση παραδείγματα προέρχονται από κάποια άγνωστη κατανομή πιθανότητας, η οποία θεωρείται αντιπροσωπευτική του χώρου των καταστάσεων, και η μηχανή πρέπει να κατασκευάσει ένα γενικό μοντέλο που θα επιτρέπει την παραγωγή προβλέψεων σε καινούργιες καταστάσεις με επαρκή ακρίβεια.

Ωστόσο η υπολογιστική ανάλυση των αλγορίθμων των μηχανών μάθησης και η απόδοσή τους είναι ένας κλάδος της θεωρητικής πληροφορικής, γνωστός ως υπολογιστική θεωρία μάθησης. Επειδή τα εκπαιδευτικά σύνολα είναι πεπερασμένα και το μέλλον αβέβαιο, η θεωρία μάθησης δεν εγγυάται πάντα την απόδοση των αλγορίθμων.

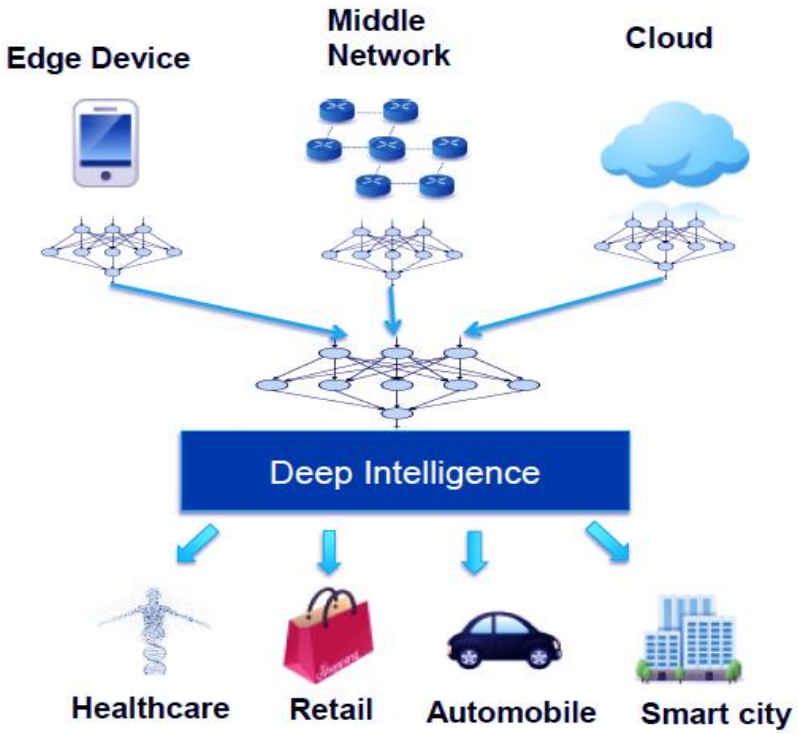
Το πόσο καλά ένα μοντέλο, που έχει εκπαιδευτεί σε υπαρκτά παραδείγματα, μπορεί να προβλέψει άγνωστες καταστάσεις ονομάζεται γενίκευση. Για την καλύτερη δυνατή γενίκευση, η πολυπλοκότητα της υπόθεσης θα πρέπει να είναι αντίστοιχη της πολυπλοκότητας της συνάρτησης των δεδομένων.

Η τεχνητή νοημοσύνη και η μηχανική μάθηση αποτελούν αναπόσπαστο μέρος του 5G. Περισσότερα δεδομένα μπορούν να μεταδίδονται προς οποιαδήποτε κατεύθυνση στο δίκτυο χωρίς να επηρεάζουν αρνητικά τη μετάδοση οποιωνδήποτε άλλων δεδομένων στο δίκτυο. Καθώς περισσότερες συσκευές έρχονται σε ένα δίκτυο 5G, καθίσταται αδύνατο να χειριστεί όλη την κίνηση που απαιτείται χωρίς τεχνητή νοημοσύνη και μηχανική μάθηση. Ένα δίκτυο 5G θα είναι σε θέση να αναλύει ιστορικά δεδομένα και συζητήσεις, επιτρέποντας έτσι την αποτελεσματικότερη μετάδοση δεδομένων ανά πάσα στιγμή. Ένα πλήρως λειτουργικό δίκτυο 5G δεν θα συμβεί χωρίς τεχνητή νοημοσύνη που μπορεί να μάθει και να λάβει αποφάσεις από μόνο του.

Όπως προαναφέρθηκε πολλές εφαρμογές του IoT χρησιμοποιούν τις μεγάλες ταχύτητες και τα πλεονεκτήματα του 5G. Οι όροι machine learning (ML) μηχανική μάθηση το Ίντερνετ των πραγμάτων (IoT) είναι δύο δημοφιλείς εκφράσεις προς το παρόν και είναι και οι δύο κοντά. Ωστόσο η Μηχανική Μάθηση είναι υποπεδίο της επιστήμης των υπολογιστών που αναπτύχθηκε από τη μελέτη της αναγνώρισης προτύπων και της υπολογιστικής θεωρίας μάθησης στην τεχνητή νοημοσύνη, μέσω της μηχανικής μάθησης δίνεται η δυνατότητα αναγνώρισης ανθρωπίνων λειτουργιών από συσκευές.

Η μηχανική μάθηση είναι στενά συνδεδεμένη και συχνά συγχέεται με υπολογιστική στατιστική, ένας κλάδος, που επίσης επικεντρώνεται στην πρόβλεψη μέσω της χρήσης των υπολογιστών, έχει ισχυρούς δεσμούς με την μαθηματική βελτιστοποίηση, η οποία παρέχει μεθόδους, τη θεωρία και τομείς εφαρμογής. Η Μηχανική μάθηση εφαρμόζεται σε μια σειρά από υπολογιστικές εργασίες, όπου τόσο ο σχεδιασμός όσο και ο ρητός προγραμματισμός των αλγορίθμων είναι ανέφικτος. Ο βασικός στόχος ενός μαθητευόμενου είναι να γενικεύει την εμπειρία του.

Σε αυτό το πλαίσιο γενίκευση είναι η ικανότητα μιας μηχανής μάθησης να αποδίδει με ακρίβεια σε καινούριες, πρωτόγνωρες εργασίες, αφού πρώτα έχει εκπαιδευτεί σε ένα σύνολο δεδομένων εκπαίδευσης. Παραδείγματα εφαρμογών αποτελούν τα φίλτρα spam (spam filtering), η οπτική αναγνώριση χαρακτήρων (OCR),οι μηχανές αναζήτησης και η υπολογιστική όραση, αναγνώριση αντικειμένων, αναγνώριση γραφικού χαρακτήρα. Επομένως η συγκέντρωση όλων αυτών των δεδομένων για ένα πείραμα σε Servers με μεγάλη υπολογιστική ισχύος και μέσα από δίκτυα υψηλών ταχυτήτων δίνουν μεγάλα ποσοστά ακρίβειας των αντικειμένων. Στην παρακάτω εικόνα αποτυπώνεται μια πρακτική αυτής της αρχιτεκτονικής .



Εικόνα 1-1: Deep Intelligence[1]

Στα επόμενα κεφάλαια θα αναπτυχθεί πως αυτές οι τεχνολογίες συμμετέχουν στο πείραμα που εφαρμόστηκε και ποια υποδομή χρησιμοποιήθηκε.

[1] https://on-demand.gputechconf.com/gtc/2015/presentation/S5813-Nobuyuki-Ota.pdf

1.1.3 Εισαγωγή στην έννοια της Αναγνώρισης Αντικειμένων

Τις δύο πρώτες δεκαετίες της έρευνας πάνω στο πεδίο της αναγνώρισης αντικειμένων υπήρχαν δύο κυρίαρχα ρεύματα, αντίστοιχα της άνωθεν (bottom-up) και της κάτωθεν (top-down) προσέγγιση της όρασης. Τα κάτωθεν μοντέλα θεωρούν ότι μια εικόνα μπορεί να αναλυθεί σε πρωτογενή σχήματα (κύλινδροι, σφαίρες) και αναζητούν τρόπους για την σύνδεση των ανιχνεύσιμων σχημάτων μέσω περιπλοκότερων μοντέλων για τα αντικείμενα. Τα άνωθεν μοντέλα χρησιμοποιώντας τρισδιάστατες αναπαραστάσεις των αντικειμένων, ανάγουν το πρόβλημα της αναγνώρισης στην εκτίμηση των παραμέτρων της προβολής του μοντέλου στην εικόνα. Στις αρχές της προηγούμενης δεκαετίας, καθώς οι περιορισμοί των παραπάνω προσεγγίσεων έγιναν αντιληπτοί, ξεκίνησαν οι πρώτες προσπάθειες αφενός για να ληφθεί υπόψη η ποικιλία της εμφάνισης που ενδέχεται να παρουσιάζουν τα αντικείμενα μια κατηγορίας και αφετέρου για την κατασκευή τέτοιων μοντέλων με μικρότερη ανθρώπινη παρέμβαση.

Σύντομα η αναγνώριση αντικειμένων στράφηκε προς τα στατιστικά μοντέλα για αναπαράσταση αντικειμένων, αντίστοιχα αυτών που έχουν επικρατήσει στην τεχνολογία φωνής και τεχνικές εκμάθησης μηχανών για την αυτόματη κατασκευή τους από δεδομένα εκπαίδευσης. Οι τεχνικές αναγνώρισης αντικειμένων μπορούν να ταξινομηθούν βάση του εύρους του προβλήματος που καλύπτουν, των μοντέλων των αντικειμένων που χρησιμοποιούν και των αλγοριθμικών τεχνικών στις οποίες βασίζονται [40][41][42].

1.1.3.1 Αναγνώριση μεμονωμένων αντικειμένων μέσω μοντέλων και αντιστοίχισης

Βάση αυτής της προσέγγισης, τα αντικείμενα που βρίσκονται σε μια δισδιάστατη εικόνα αναγνωρίζονται χρησιμοποιώντας αποθηκευμένες αναπαραστάσεις τρισδιάστατων αντικειμένων ή συνδυασμούς δισδιάστατων όψεων τους. Οι βασικές συνιστώσες του προβλήματος είναι η κατασκευή των αναπαραστάσεων, η παραγωγή ενός περιορισμένου πλήθους υποθέσεων δεδομένης μιας εικόνας και η εκτίμηση της πιστότητας των παρατηρήσεων στις προβλέψεις του μοντέλου. Για την αναζήτηση της θέσης του αντικειμένου μπορούν να χρησιμοποιηθούν άνωθεν τεχνικές ή κάτωθεν τεχνικές όπως ο μετασχηματισμός Hough. Έχοντας κάποιες υποψήφιες τοποθεσίες, η εκτίμηση της πιστότητας των παρατηρήσεων επιτυγχάνεται βάσει κάποιου κατάλληλου μέτρου της απόστασης μεταξύ των χαρακτηριστικών των παρατηρήσεων και του μοντέλου. Αν και οι προσεγγίσεις αυτές έχουν δώσει ικανοποιητικά αποτελέσματα σε πρακτικές εφαρμογές (π.χ. επισκόπηση εξαρτημάτων σε βιομηχανικές γραμμές συναρμολόγησης, ανάκληση αντικειμένων από μία βάση δεδομένων), το πεδίο εφαρμογών τους είναι περιορισμένο σε συγκεκριμένα αντικείμενα αντί για κατηγορίες αντικειμένων. Όμως, η εμπειρία που έχει

συσσωρευτεί έχει αποδειχθεί ωφέλιμη για την διατύπωση και επίλυση νέων προβλημάτων που προκύπτουν σε σύγχρονες τεχνικές αναγνώρισης αντικειμένων.

1.1.3.2 Αναγνώριση αντικειμένων με τεχνικές αναγνώρισης προτύπων & νευρωνικών δικτύων

Οι τεχνικές αυτές ξεκίνησαν από την έρευνα στο χώρο των νευρωνικών δικτύων, όπου το πρόβλημα της ανίχνευσης αντιμετωπίζεται μέσω της παράλληλης κατανεμημένης επεξεργασίας της πληροφορίας. Ξεκινώντας από την εξαγωγή ενός συνόλου χαρακτηριστικών, τα οποία είναι αναλλοίωτα ως προς ανεπιθύμητες πηγές ποικιλίας στην εμφάνιση, όπως ο φωτισμός, η ανίχνευση ενός αντικειμένου, εκφράζεται ως ένα πρόβλημα αναγνώρισης προτύπων. Με άλλα λόγια δεδομένου ενός συνόλου χαρακτηριστικών μέσα στο εξεταζόμενο πλαίσιο, ζητείται να ληφθεί η απόφαση για το εάν υπάρχει ένα αντικείμενο. Τα προβλήματα που καλούνται να αντιμετωπίσουν τέτοιες τεχνικές, αφορούν την εξαγωγή κατάλληλων χαρακτηριστικών για τα αντικείμενα, την αποδοτική υλοποίηση των αλγορίθμων ανίχνευσης καθώς και όλα τα προβλήματα του πεδίου της αναγνώρισης προτύπων. Από τα βασικά πλεονεκτήματα των μεθόδων αυτών θεωρούνται η ωριμότητα των διαθέσιμων τεχνικών από την περιοχή της μηχανικής μάθησης (SVM) καθώς και η ταχύτητα με την οποία μπορεί να γίνει η διαδικασία της ανίχνευσης. Ωστόσο υπάρχουν δυο κύρια προβλήματα. Το βασικότερο είναι ότι για την εκμάθηση τους χρειάζεται συνήθως ένα σύνολο εκπαίδευσης της τάξης των μερικών χιλιάδων εικόνων, καθώς περιλαμβάνεται ένα μεγάλο πλήθος παραμέτρων που πρέπει να εκμαθευτούν. Αντίθετα εμείς ως άνθρωποι αρκούμαστε σε μια εικόνα για να επιτύχουμε ικανοποιητικά αποτελέσματα. Επίσης μια τέτοια προσέγγιση δεν είναι εύκολο να αντιμετωπίσει επικαλύψεις με άλλα αντικείμενα και γενικότερα προβλήματα που δεν συμπεριλαμβάνονται στο σύνολο εκπαίδευσης.

1.1.3.3 Αναγνώριση αντικειμένων με τμηματικές αναπαραστάσεις

Τα μοντέλα που ανήκουν στην κατηγορία των τμηματικών ταξινομητών βασίζουν την ανίχνευση των αντικειμένων τους στην ανίχνευση χαρακτηριστικών σημείων (keypoints) των αντικειμένων. Θεωρείται εύκολο να χρησιμοποιηθεί στατιστική πληροφορία για τη χωρική διάταξη των τμημάτων, ώστε να εισαχθεί και γεωμετρική γνώση στη διαδικασία ανίχνευσης. Η προσέγγιση αυτή συνδυάζει την αποδοτικότητα του κάτωθεν μοντέλου της όρασης, στο οποίο από επιμέρους τμήματα αναγνωρίζονται περιπλοκότερα αντικείμενα, με τους περιορισμούς που εισάγει η πρότερη γνώση των άνωθεν μοντέλων απλοποιώντας την αναζήτηση σε συνδυασμούς που θα μπορούσαν να έχουν προκύψει από το αντικείμενο. Το βασικό της μειονέκτημα είναι ότι αντί για το αντικείμενο κάθε αυτό μοντελοποιούνται τα τμήματα του και οι αλληλεξαρτήσεις τους που θεωρούνται χρήσιμες για την ανίχνευση τους, παρέχοντας έτσι ένα μοντέλο περιορισμένων δυνατοτήτων για άλλες λειτουργίες, όπως για παράδειγμα την αναγνώριση της ταυτότητας ενός αντικειμένου.

1.1.3.4 Αναγνώριση αντικειμένων με παραμορφώσιμα μοντέλα

Αυτή η προσέγγιση βασίζεται στην χρήση ενός παραμετρικού μοντέλου των παραμορφώσεων της εμφάνισης και γεωμετρίας ενός πρωτότυπου αντικειμένου, το οποίο καλύπτει όλη την ποικιλία που χαρακτηρίζει μια κατηγορία αντικειμένων. Δεδομένης μίας νέας εικόνας και μίας αρχικοποίησης, το πρόβλημα της αναγνώρισης/ανίχνευσης ανάγεται στην μεταβολή των παραμέτρων του μοντέλου ώστε αυτό να αναπαράγει τις παρατηρήσεις. Οι δύο βασικές συνιστώσες της προσέγγισης αυτής είναι η μοντελοποίηση της παραμόρφωσης και η εκτίμηση των παραμέτρων των μοντέλων. Ως πρώτες προσπάθειες στην κατεύθυνση της μοντελοποίησης της παραμόρφωσης του σχήματος μπορούν να θεωρηθούν αφενός τεχνικές όπως τα παραμορφώσιμα πρότυπα (Deformable Templates) αφετέρου η αναγνώριση μέσω συνδυασμού των όψεων. Τα μοντέλα αυτά θεωρούν ότι οι παρατηρήσεις που προέρχονται από το αντικείμενο μπορούν να μοντελοποιηθούν από την χωρική παραμόρφωση του αντικειμένου ακολουθουμένη από την παραμόρφωση της φωτεινότητας του. Το βασικό πρόβλημα με αυτά τα μοντέλα είναι ότι καθώς χρησιμοποιείται ένας επαναληπτικός αλγόριθμος δεν είναι εφικτό να ερευνήσει κανείς για όλες τις θέσεις, κλίμακες τις εικόνας. Συνεπώς τέτοιες μέθοδοι δεν είναι κατάλληλες για την εξαρχής ανίχνευσης αντικειμένων, αλλά μπορούν να χρησιμοποιηθούν για τη λήψη της τελικής απόφασης. Για αυτό χρησιμοποιούνται κυρίως για αναγνώριση αντικειμένων θεωρώντας γνωστό ότι στη θέση που αρχικοποιούνται υπάρχει ένα αντικείμενο κάποιας συγκεκριμένης κατηγορίας. Μια άλλη εφαρμογή τους είναι η καταγραφή της εξέλιξης

αντικειμένων (tracking), κατά την οποία χρησιμοποιείται η πρόβλεψη βάσει της προηγούμενης θέσης του μοντέλου για να ξεκινήσει η αναζήτηση της τρέχουσας.

1.1.4 Υπολογιστική Νέφους

Σύμφωνα με τα παραπάνω παρατηρείται στην έναρξη κιόλας αυτών των τεχνολογιών μεγάλη απαίτηση σε υπολογιστική ισχύ σε ένα πολύ σημαντικό και επιταχυνόμενο ρυθμό. Επιπρόσθετα, θα εμφανιστούν στον ορίζοντα νέες εφαρμογές όπως αυτές τις επαυξημένης πραγματικότητας (Augmented Reality) AR[46], αναγνώρισης αντικειμένων, δυναμική αλληλεπίδραση μεταξύ αντικειμένων (π χ αυτόνομα οχήματα) όπου όλα αυτά σηματοδοτούν την ανάγκη για την ανάπτυξη του Mobile Cloud Computing (MCC)[44][45]. Ωστόσο το MCC επιβάλλει πρόσθετες ανάγκες και φορτίο τόσο σε ραδιοφωνικά όσο και σε ευρυζωνικά δίκτυα, λόγω της υψηλής και μεταβλητής καθυστέρησης σε απομακρυσμένα κέντρα δεδομένων. Με βάση αυτά τα δεδομένα και ευρήματα, το Ευρωπαϊκό Ινστιτούτο Τηλεπικοινωνιών και προτύπων- European Telecommunications Standard Institute (ETSI) εισήγαγε μια νέα έννοια, την πολλαπλά προσπελάσιμη Multi-access Edge Computing (MEC)[42][43]. Στην ουσία, η MEC αξιοποιεί την υπολογιστική ενός network edge (π.χ. διακομιστές ή άλλους υπολογιστικούς πόρους κοντά σε basestations, σημεία πρόσβασης Wifi ή ευρυζωνικών δικτύων).

Το MEC είναι ένα βασικό συστατικό στοιχείο για την υλοποίηση των εφαρμογών του Internet of Things (IoT) σε περιβάλλοντα έξυπνων πόλεων. Η MEC προσφέρει πολλά πλεονεκτήματα στους τελικούς χρήστες:

i) πρόσβαση σε ισχυρά στοιχεία υπολογιστών
ii) μείωση της κατανάλωσης ενέργειας των κινητών συσκευών
iii) χαμηλότερη καθυστέρηση δικτύου
iv) υψηλότερες επιδόσεις στα μέρη των εφαρμογών που αναπτύσσονται σε edge cloud.

1.1.5 Το νέο δίκτυο 5G

Τα νέα δίκτυα επικοινωνιών, το περίφημο 5G, βρίσκονται προ των πυλών και θα φέρουν μια πραγματική επανάσταση στον τρόπο ζωής που γνωρίζουμε μέχρι σήμερα, όχι μόνον στον τομέα της οικονομίας, αλλά και στην υγεία, τις μεταφορές, την ψυχαγωγία και φυσικά στην καθημερινότητα.

Ο αριθμός των Ευρωπαίων παρόχων που επενδύουν σε τεχνολογίες 5G αυξάνεται σημαντικά, καθώς η βιομηχανία τηλεπικοινωνιών οδεύει με ταχείς ρυθμούς προς την εμπορική

εισαγωγή των τεχνολογιών 5G. Οι κύριοι πάροχοι στην Ευρώπη έχουν ήδη ανακοινώσει τα πρώτα αποτελέσματα των δοκιμών και τα σχέδια για περαιτέρω παρουσιάσεις συγκεκριμένων χαρακτηριστικών του 5G. Το πανευρωπαϊκό πρόγραμμα «Στρατηγική 5G» έχει ήδη ξεκινήσει σε αρκετές χώρες, οδηγώντας το οικοσύστημα στην επίτευξη των αναπτυξιακών ορόσημων του 5G, όπως καθορίζονται από την Ευρωπαϊκή Επιτροπή.

Αυτά, συγκεκριμένα, περιλαμβάνουν: μέχρι το 2018 να είχε καθοριστεί η στρατηγική 5G για κάθε χώρα της ΕΕ, μέχρι το 2020 να ξεκινήσει η εμπορική χρήση δικτύου 5G σε μία επιλεγμένη πόλη σε κάθε χώρα-μέλος και τέλος, μέχρι το 2025, να έχουμε πλήρως λειτουργικό δίκτυο 5G που θα καλύπτει όλες τις αστικές περιοχές και τις μεγάλες χερσαίες αρτηρίες των μεταφορών.

Ως εκ τούτου, το αργότερο μέχρι το 2025, όλες οι Ευρωπαϊκές χώρες απαιτείται να έχουν διαθέσιμη εκτενή κάλυψη 5G, ανάφερε ο Dr Jakub Borkowski, Wireless Business Development Director for CEE & Nordic της Huawei.

1.1.5.1 Χαρακτηριστικά 5G

Το χιλιοστό κύμα 5G είναι το ταχύτερο, με πραγματικές ταχύτητες που συχνά είναι 1-2 Gbit / s κάτω. Οι συχνότητες είναι πάνω από 24 GHz και φθάνουν μέχρι τα 72 GHz και βρίσκονται πάνω από τα κατώτατα όρια της ζώνης εξαιρετικά υψηλής συχνότητας. Η εμβέλεια είναι μικρή, έτσι απαιτούνται περισσότερα κελιά.

Η μεσαία ζώνη 5G είναι η πιο ευρέως αναπτυγμένη, σε περισσότερα από 20 δίκτυα. Οι ταχύτητες σε μια ευρεία ζώνη των 100 MHz είναι συνήθως 100-400 Mbit / s κάτω. Στο εργαστήριο και περιστασιακά στο πεδίο, οι ταχύτητες μπορούν να υπερβούν ένα gigabit ανά δευτερόλεπτο. Οι συχνότητες που αναπτύσσονται είναι από 2,4 GHz έως 4,2 GHz.

1.1.5.2 Ταχύτητα

Οι ταχύτητες 5G κυμαίνονται από ~ 50Mbit / s έως πάνω από 2 gigabit στην αρχή και αναμένεται να φτάσουν ακόμα και 100Gbit / s, 100x ταχύτερα από 4g. Η ταχύτερη 5g, γνωστή ως mmWave, προσφέρει ταχύτητες μέχρι και πάνω από 2Gbit / s. Το πρόβλημα με αυτό όμως είναι ότι δεν μπορεί να περάσει από τοίχους δέντρα, κλπ. Λόγω της υψηλής συχνότητας.

Το φάσμα χαμηλής ζώνης προσφέρει την πλησιέστερη περιοχή κάλυψης, αλλά είναι πιο αργή από τις άλλες, αν και ακόμα ταχύτερη από 4g.

Η ταχύτητα 5G NR στις ζώνες κάτω των 6 GHz μπορεί να είναι ελαφρώς υψηλότερη από την 4G με παρόμοιο μέγεθος φάσματος και κεραίες , αν και ορισμένα δίκτυα 3GPP 5G θα

είναι πιο αργά από κάποια προηγμένα δίκτυα 4G.. Η προδιαγραφή 5G επιτρέπει επίσης LAA (Assisted Accessed Access), αλλά η LAA στο 5G δεν έχει ακόμη αποδειχθεί. Η προσθήκη LAA σε υπάρχουσα διαμόρφωση 4G μπορεί να προσθέσει εκατοντάδες megabits ανά δευτερόλεπτο στην ταχύτητα, αλλά αυτή είναι μια επέκταση του 4G, όχι ένα νέο μέρος του προτύπου 5G.

Η ομοιότητα σε όρους απόδοσης μεταξύ 4G και 5G στις υπάρχουσες ζώνες είναι επειδή η 4G πλησιάζει ήδη το όριο Shannon για τα ποσοστά επικοινωνίας δεδομένων. Οι ταχύτητες 5G στο λιγότερο κοινό φάσμα χιλιοστομετρικών κυμάτων, με το πολύ πιο άφθονο εύρος ζώνης και μικρότερη εμβέλεια και συνεπώς μεγαλύτερη δυνατότητα επαναχρησιμοποίησης συχνότητας, μπορεί να είναι σημαντικά υψηλότερες[47].

1.1.5.3 Καθυστέρηση

Στο 5G, η "καθυστέρηση" το 2019 ήταν 8-12 χιλιοστά του δευτερολέπτου. Η καθυστέρηση στον διακομιστή πρέπει να προστεθεί στην "καθυστέρηση αέρα" air letency. Κάποιοι πάροχοι αναφέρουν ότι η λανθάνουσα κατάσταση στην αρχική της ανάπτυξη 5G είναι 30 ms. Οι διακομιστές Edge κοντά στους πύργους μπορούν να μειώσουν την καθυστέρηση σε 10-20 ms. Τα 1-4 ms θα είναι εξαιρετικά σπάνια εκτός εργαστηρίων. Επίσης, η προσομοίωση δικτύου μπορεί να χρησιμοποιηθεί για την πρόβλεψη της απόδοσης των δικτύων 5G πριν από την ανάπτυξη[48].

1.1.5.4 Ευρυζωνικές ταχύτητες

Η τεχνολογία 5G υπόσχεται ένα πρωτοφανές άλμα στις ευρυζωνικές ταχύτητες, συγκριτικά με τα προηγούμενα δίκτυα κινητής τηλεφωνίας. Η 5G είναι η ασύρματη τεχνολογία πέμπτης γενιάς για ψηφιακά κυψελοειδή δίκτυα που άρχισαν να αναπτύσσονται ευρέως το 2019. Όπως και με τα προηγούμενα πρότυπα, οι περιοχές με κάλυψη χωρίζονται σε περιοχές που ονομάζονται "κελιά" και εξυπηρετούνται από μεμονωμένες κεραίες. Σχεδόν κάθε σημαντικός πάροχος τηλεπικοινωνιακών υπηρεσιών στον ανεπτυγμένο κόσμο αναπτύσσει κεραίες ή προτίθεται να τις αναπτύξει σύντομα. Το φάσμα συχνοτήτων του 5G χωρίζεται σε χιλιοστά κύματα, μεσαία ζώνη και χαμηλή ζώνη. Η χαμηλή ζώνη χρησιμοποιεί παρόμοια περιοχή συχνοτήτων με τον προκάτοχό της, 4G.

Η τεχνολογία 5G θα μειώσει τη χρονοκαθυστέρηση και θα βελτιώσει τη συνολική απόκριση του δικτύου. Επιπλέον, βελτιώνοντας την αρχιτεκτονική του δικτύου θα επιτευχθεί η απαίτηση για συνολική χρονοκαθυστέρηση, μικρότερη από 5ms.

Το πιο σημαντικό χαρακτηριστικό των νέων δικτύων 5G είναι η λειτουργία Ultra - reliable low latency Communication (URLLC) που θα θέσει ισχυρά θεμέλια για αυτοματοποίηση εφαρμογών μεταξύ διαφορετικών κλάδων, υπηρεσίες από μηχάνημα σε μηχάνημα και εφαρμογές δημόσιας ασφάλειας. Όλες οι προηγούμενες τεχνολογίες κινητής επικοινωνίας αδυνατούσαν να παρέχουν αξιόπιστη συνδεσιμότητα, που αποτελεί μια κρίσιμη απαίτηση για τη συντριπτική πλειονότητα των βιομηχανικών εφαρμογών.

Αυτό θα βοηθήσει τις εταιρείες να μειώσουν τα λειτουργικά τους έξοδα, να βελτιώσουν τη δραστηριότητά τους και, ταυτόχρονα, θα κάνει εφικτές πληθώρα ρηξικέλευθων εφαρμογών.

Ενδεικτικά αναφέρουμε: τα αυτόνομα οχήματα, συνδεδεμένα drones για παραδόσεις προϊόντων, υπηρεσίες τηλεϊατρικής, και πολλά άλλα. Με αυτόν τον τρόπο, οι καινοτομίες 5G θεωρούνται ότι οδηγούν στην οικονομική και κοινωνική ανάπτυξη με εντελώς νέους τρόπους.

Συγκρίνοντας τα υπάρχοντα συστήματα κινητών τηλεφώνων, το 5G είναι το πρώτο που διαχειρίζεται απευθείας την ψηφιοποίηση και την αυτοματοποίηση των υπηρεσιών για τους καταναλωτές. Ως εκ τούτου αναμένεται ότι το 5G θα φέρει επανάσταση στον τρόπο ζωής μας και θα επηρεάσει, μακροπρόθεσμα, τον τρόπο που αντιλαμβανόμαστε την πραγματικότητα.

Με το 5G ο χρήστης θα έχει συνεχώς στη διάθεσή του online, υψηλής ποιότητας, cloud-based περιεχόμενο που περιλαμβάνει: παιχνίδια, εφαρμογές VR/AR, βίντεο με ανάλυση 4k/8k και πολλά άλλα. Το 5G smartphone θα λειτουργεί ως πύλη διασύνδεσης με το δίκτυο και σαν ένα τερματικό συνδεδεμένο, μέσω εξαιρετικά υψηλών ταχυτήτων και με σχεδόν απεριόριστους πόρους στο διαδικτυακό νέφος (cloud).

Μπορούμε επίσης να αναμένουμε με βεβαιότητα, ότι τέτοια τεχνολογική διαθεσιμότητα και αρμονική συνύπαρξη με το cloud, θα οδηγήσει την ανάπτυξη καινοτόμων εφαρμογών που θα αλλάξουν πλήρως τον σημερινό τρόπο χρήσης του διαδικτύου και των ψηφιακών τεχνολογιών.

1.1.5.5 Εφαρμογές 5G

Σταδιακά, το δίκτυο 5G θα συνεισφέρει, ώστε να αναπτυχθούν πολλές εφαρμογές που θα λειτουργούν με μεγαλύτερη ακρίβεια και μικρότερο κόστος. Για παράδειγμα, οι έξυπνες πόλεις και τα έξυπνα σπίτια θα φροντίζουν την ποιότητα ζωής μας, η υγεία μας θα βρίσκεται υπό τη διαρκή φροντίδα των υπολογιστών που θα επεξεργάζονται σήματα από αισθητήρες στο περιβάλλον μας, δέματα θα παραδίδονται με drones και η δημόσια ασφάλεια θα παρέχεται από διασυνδεδεμένες υπηρεσίες γρήγορης ανταπόκρισης, με δυνατότητα αποτελεσματικότερης δράσης χάρις στη συνεχή διαθεσιμότητα όλων των απαραίτητων πληροφοριών.

Επιπροσθέτως, θα χρησιμοποιούμε αυτόνομα Μέσα Μαζικής Μεταφοράς και αυτοκίνητα, που θεωρείται μια πραγματική επανάσταση που επηρεάζει σημαντικά, με θετικό αντίκτυπο, τον τρόπο ζωής συνδεδεμένα αυτοκίνητα, έξυπνη παραγωγή, η συνδεδεμένη

ενέργεια, η e-υγεία, η οικιακή ψυχαγωγία, έξυπνες διαδικασίες βιομηχανικής παραγωγής, τα συνδεδεμένα drones, οι έξυπνες πόλεις καθώς και πολλές άλλες εφαρμογές σε τομείς όπου αναμένεται ένα πρώτο κύμα των 5G καινοτομιών.

Αυτά τα χαρακτηριστικά είναι θεμελιώδη για την παροχή της απαραίτητης απόδοσης σε υψηλές ταχύτητες και σε πυκνοκατοικημένες περιοχές και μόνο η τεχνολογία 5G μπορεί να ικανοποιήσει αυτές τις αυστηρές απαιτήσεις συνδεσιμότητας.

1.1.5.6 Βιομηχανική Αυτοματοποίηση

Το 5G αναμένεται να είναι καταλύτης για τη βιομηχανική αυτοματοποίηση. Η καινοτομία είναι η καρδιά της παραγωγής. Μείζονες εξελίξεις περιλαμβάνουν ενέργειες που οδηγούν σε αποδοτικές και λιτές παραγωγικές διαδικασίες, ψηφιοποίηση και μεγαλύτερη ευελιξία στην διεκπεραίωση εργασιών καθώς και στην παραγωγή. Η υποκείμενη επιχειρηματική λογική για την υλοποίηση της έξυπνης παραγωγής είναι η παράδοση υψηλότερης ποιότητας προϊόντων στην αγορά γρηγορότερα, με περισσότερο ευέλικτα και αποδοτικά συστήματα παραγωγής.

Πρόσφατα έχει παρατηρηθεί μία έντονη τάση που ευνοεί τον τομέα του Internet of Things (IoT). Ιστορικά, οι κατασκευαστές βασίζονταν στις ενσύρματες τεχνολογίες για τις συνδεδεμένες τους εφαρμογές. Ωστόσο, οι ασύρματες λύσεις, όπως το Wi- Fi και το Bluetooth, πήραν πρόσφατα το προβάδισμα στον παραγωγικό κλάδο, όμως αυτές οι ασύρματες λύσεις αντιμετωπίζουν περιορισμούς στην ασφάλεια και την αξιοπιστία της ευρυζωνικής διασύνδεσης. Στο σημείο αυτό η τεχνολογία 5G αναμένεται να επιφέρει πολύ σημαντικές αλλαγές και εξελίξεις.

1.1.6 Το 5G και Διαδικτύου των Πραγμάτων

Το Διαδίκτυο των πραγμάτων (IoT) είναι ένα σύστημα αλληλένδετων υπολογιστικών συσκευών, οι μηχανικές και ψηφιακές μηχανές παρέχονται με μοναδικά αναγνωριστικά (UID) και τη δυνατότητα μεταφοράς δεδομένων μέσω δικτύου χωρίς να απαιτείται αλληλεπίδραση ανθρώπου με άνθρωπο ή άνθρωπος-προς-υπολογιστή.

Πιο απλά, αποτελεί το δίκτυο επικοινωνίας πληθώρας συσκευών, οικιακών συσκευών, αυτοκινήτων καθώς και κάθε αντικειμένου που ενσωματώνει ηλεκτρονικά μέσα, λογισμικό, αισθητήρες και συνδεσιμότητα σε δίκτυο ώστε να επιτρέπεται η σύνδεση και η ανταλλαγή δεδομένων. Απλούστερα, η φιλοσοφία του IoT είναι η σύνδεση όλων των ηλεκτρονικών

συσκευών μεταξύ τους (τοπικό δίκτυο) ή με δυνατότητα σύνδεσης στο διαδίκτυο (παγκόσμιο ιστό).

Παραδείγματα (IoT) που εστιάζουν στην έξυπνη ζωή έχουν εφαρμοστεί με επιτυχία. Ο δήμος των ΗΠΑ που πέρασε σε έξυπνους μετρητές για την παρακολούθηση της χρήσης του νερού είδε άμεση και ασφαλή εξοικονόμηση χρημάτων. Η διαδικασία συλλογής δεδομένων εξελίχθηκε από μια χειρωνακτική διαδικασία, στην οποία οι τεχνικοί ταξίδευαν σε κάθε μετρητή, σε μία γρήγορη και τυποποιημένη εργασία από συστήματα IoT όπου οι αυτόματοι μετρητές κατέγραφαν και απέστειλαν σε μια κεντρική βάση δεδομένων τα αποτελέσματα των μετρήσεων.

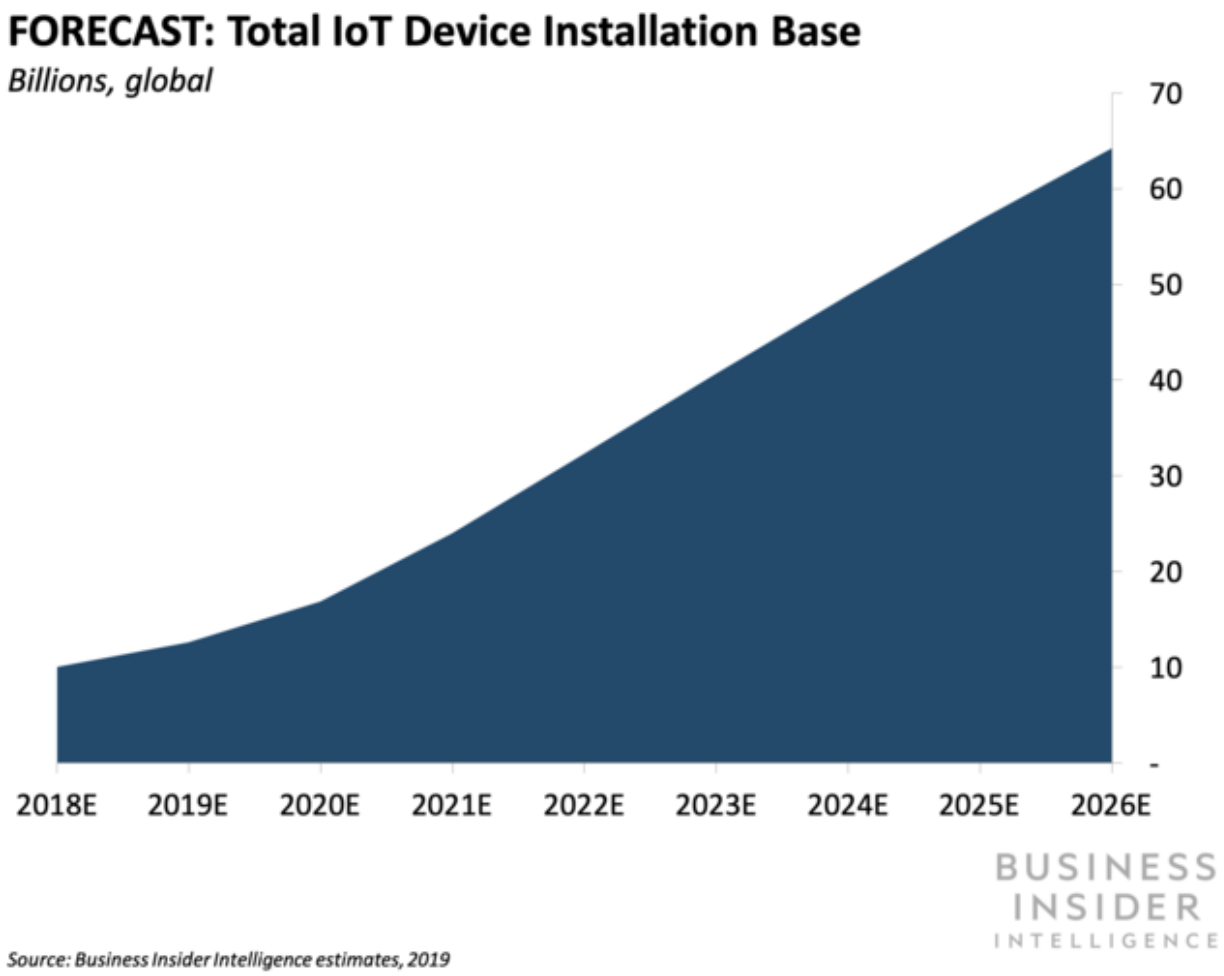
Βασισμένη στο IoT πρόσφατα παρουσιάστηκε ο αρχιτεκτονικός σχεδιασμός και η πειραματική αξιολόγηση μιας υπηρεσίας που υποστηρίζει το IoT για έξυπνες τουριστικές περιοχές, αξιοποιώντας τις λειτουργίες NFV/SDN μιας πειραματικής υποδομής μεγάλης κλίμακας. Ειδικότερα αξιολογήθηκε η απόδοση μιας υπηρεσίας δικτύου που λαμβάνει εικόνες από ένα πλήθος σημείων ενδιαφέροντος (PoIs), εκτελεί τον προσδιορισμός αντικειμένου χρησιμοποιώντας ένα εκπαιδευμένο μοντέλο, υπολογίζει την ακρίβεια πρόβλεψης και επιστρέφει στους τελικούς χρήστες το αποτέλεσμα ταυτοποίησης με ακρίβεια μαζί με χρήσιμες πληροφορίες για το αναγνωρισμένο αντικείμενο.

Ήδη σήμερα, οι υπηρεσίες IoT αναδεικνύονται σταδιακά σε μια σημαντική πηγή εσόδων για τους παρόχους των ασύρματων δικτύων, χωρίς ωστόσο να είναι ακόμα η βασικότερη. Οι συμβατικές τεχνολογίες δικτύων, ακόμη και το LTE, δεν είναι σε θέση να διαχειριστούν εκατομμύρια συνδεδεμένων συσκευών ανά τετραγωνικό χιλιόμετρο.

Ο ρόλος του 5G είναι να διαχειριστεί την αγορά του IoT, όσον αφορά στην παροχή υψηλότερου επιπέδου ασφάλειας και σημαντικά αυξημένης χωρητικότητας, δηλαδή μεγαλύτερος αριθμός ταυτόχρονα συνδεδεμένων αισθητήρων. Ως εκ τούτου, το 5G θεωρείται ένα ισχυρό θεμέλιο για την ανάπτυξη του IoT.

Τεχνικά, τα πρότυπα υποστήριξης των υπηρεσιών τύπου IoT από το 5G αναμένεται να έχουν ολοκληρωθεί μέχρι το 2020 από τους οργανισμούς Τυποποίησης και, μέχρι τότε, οι διαχειριστές δικτύων θα χρησιμοποιούν την εξελιγμένη λειτουργία του LTE, που ονομάζεται NB-IOT (Narrowband-IoT) η οποία όμως υστερεί σημαντικά σε σχέση με τις δυνατότητες που θα παρέχει στο μέλλον η τεχνολογία 5G. Η Business Insider Intelligence αναμένει ότι θα εγκατασταθούν περισσότερες από 64 δισεκατομμύρια συσκευές IoT σε όλο τον κόσμο μέχρι το 2026.[2]

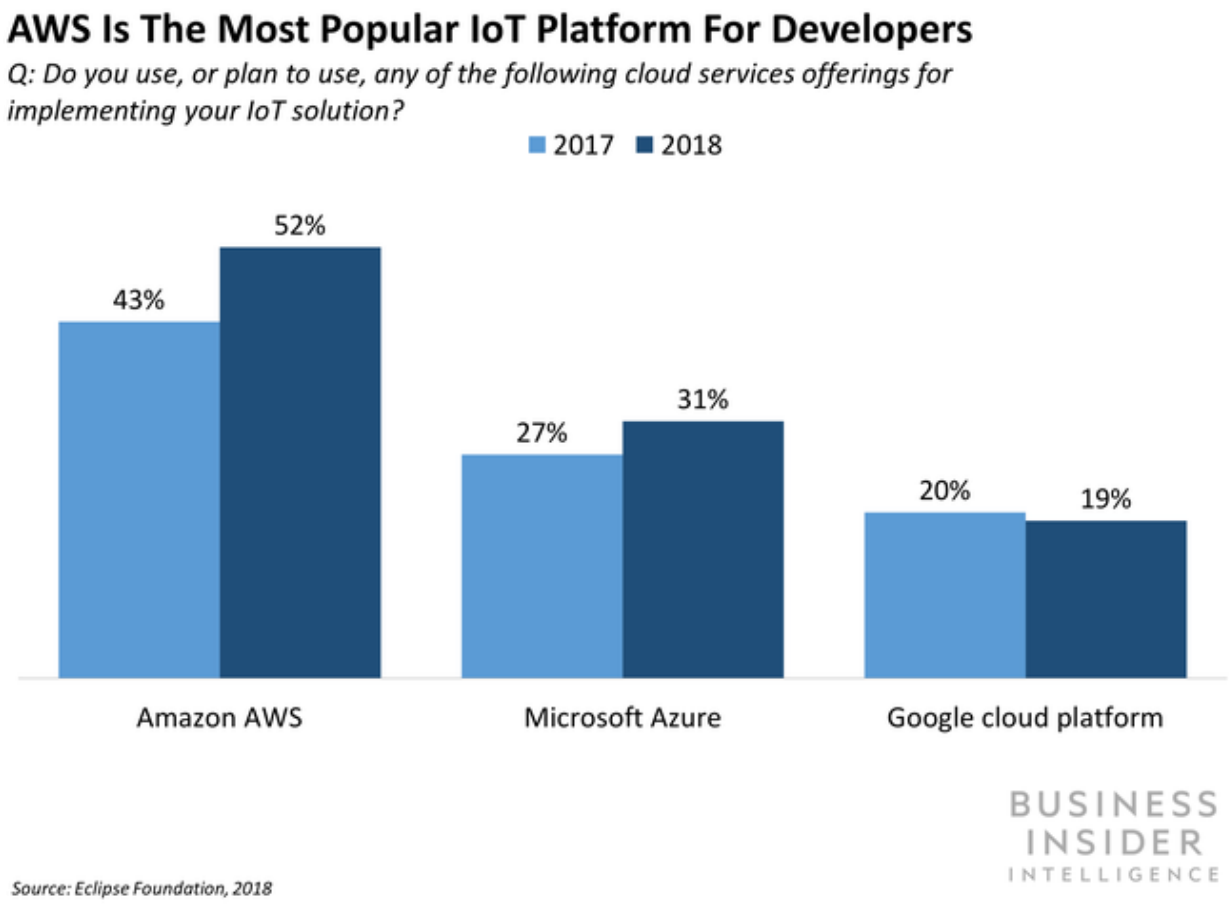
[22] https://www.businessinsider.com/iot-infrastructure-technology?r=US&IR=T



Εικόνα 1-2:IoT Device Installation[3]

Μερικές από τις πιο δημοφιλείς πλατφόρμες cloud της IoT στην αγορά περιλαμβάνουν τις υπηρεσίες Amazon Web Services, GE Predix, Google Cloud IoT, Microsoft Azure IoT Suite, IBM Watson και Salesforce IoT Cloud[49].

[3] https://www.businessinsider.com/iot-infrastructure-technology?r=US&IR=T



Εικόνα 1-3:Popular IoT Platform [4]

1.2 Σκοπός και Συνεισφορά Διατριβής

Σκοπός της παρούσας εργασίας, είναι η ανάπτυξη και η καταγραφή μίας πειραματικής μελέτης που συνδυάζει τις παραπάνω τεχνολογίες αφορά αναγνώριση αντικειμένων από φορητές συσκευές με τη χρήση της υπολογιστικής νέφους. Η υποδομή του πειράματος βασίζεται σε υπηρεσίες αναγνώρισης αντικειμένων οι οποίες είναι εγκατεστημένες σε Server.Το πείραμα εκτελείται με την εξής διαδικασία, πραγματοποιεί τη λήψη εικόνων από εφαρμογή εγκατεστημένη σε φορητή συσκευή, πραγματοποιείται η αποστολή σε εικονικό server ο οποίος είναι προσαρμοσμένος στις ανάγκες μίας υπηρεσίας όπου εκτελείται η αναγνώριση της εικόνας, ο κεντρικός υπολογιστής λαμβάνει και αποδίδει δεδομένα χρησιμοποιώντας ένα εκπαιδευμένο μοντέλο βασισμένο στην μηχανική μάθηση (με βάση το Tensorflow), υπολογίζει την ακρίβεια

[4] https://www.businessinsider.com/iot-infrastructure-technology?r=US&IR=T

και την πρόβλεψη της εικόνας και επιστρέφει στον τελικό χρήστη της φορητής συσκευής πληροφορίες σχετικά με το αναγνωρισμένο αντικείμενο.

Η συνεισφορά της διατριβής έχει εκτός από θεωρητική προσέγγιση και πρακτική, καθώς θα δούμε τη μελέτη του συγκεκριμένου πειράματος που πραγματοποιήθηκε σε εργαστηριακό περιβάλλον. Στην διατριβή παρουσιάζονται η υποδομή των τεχνολογιών που χρησιμοποιήθηκαν, το περιβάλλον του COSMOS στο οποίο βασίστηκε η ιδέα της περεταίρω αξιολόγησης των πόρων που απαιτούνται για το εγχείρημα. Επιπρόσθετα, γίνεται μία αναλυτική περιγραφή της διαδικασίας εκπαίδευσης του μοντέλου για την ανίχνευση των εικόνων η οποία αποτελείται από την εγκατάσταση του Tensorflow το lambeling των εικόνων και της τελικής αναγνώρισης των εικόνων. Γίνεται αναφορά στην υποδομή του Server που είναι εγκατεστημένη η υπηρεσία ανίχνευσης εικόνων και που δέχεται τα αιτήματα από τη φορητή συσκευή. Τέλος, καταγράφονται οι μετρήσεις της απαιτούμενης επεξεργαστικής ισχύος CPU (Ghz), της μνήμης (MB) και του χρόνου απόκρισης του server που καλύπτουν τους παρακάτω στόχους.

Στόχοι του πειράματος είναι να καταγραφούν σε διαγράμματα οι δείκτες απόδοσης συσκευών που υπάρχουν σε δίκτυα νέφους. Ο διακομιστής που φιλοξενεί την υπηρεσία ανίχνευσης εικόνων κάνει κατανομή διαχείριση χρήσης των πόρων που διαθέτει για βέλτιστη απόδοση. Ωστόσο μέσα από το πείραμα θα ληφθούν μετρήσεις για την χρήση των συγκεκριμένων πόρων, για να επιτευχθούν οι στόχοι του πειράματος ώστε να καταγραφεί η κατανάλωση πόρων σε υπολογιστική ισχύ καθώς και χρόνος απόκρισης της ανίχνευσης της εικόνας στον διακομιστή που δέχεται τα αιτήματα έγιναν οι παρακάτω πειραματικές μετρήσεις, αποστολή στον διακομιστή τυχαίων ομάδων εικόνων με τη χρήση κώδικα αναπτυγμένο σε python, ο οποίος στέλνει τις εικόνες στον server και επιστρέφει τους υπολογιζόμενους δείκτες ανά ομάδα εικόνων. Οι καταγραφές των ληφθέντων τιμών του μέσου όρου ανά ομάδα εικόνων παρουσιάζονται σε διαγράμματα και αφορούν όπως αναφέρθηκε τους δείκτες επεξεργαστική ισχύ (CPU), μνήμη (Memory) και χρόνο απόκρισης Time Response του διακομιστή που φιλοξενεί την υπηρεσία ανίχνευσης εικόνων.

1.3 Δομή Διπλωματικής

Στην Εισαγωγή κεφάλαιο πρώτο (1ο) αναπτύσσονται αναλυτικά οι νέες τεχνολογίες αιχμής και πώς αυτές επηρεάζουν τη σύγχρονη ζωή. Η υπολογιστική νέφους και η αναγνώριση αντικειμένων στις οποίες βασίζεται το πείραμα μας. Οι εφαρμογές του 5G και πως αυτό δίνει ώθηση στις εφαρμογές Internet of Things. Επιπρόσθετα στο ίδιο κεφάλαιο αναπτύσσονται οι τεχνολογίες IoT και που βρίσκουν εφαρμογές με σημαντική αναφορά στον τομέα μηχανικής μάθησης και IoT. Ο συνδυασμός των δυο τεχνολογιών κάνει κατανοητό το πώς αυτές επηρεάζουν την σύγχρονη ζωή και οδηγούν σε έξυπνα σπίτια και έξυπνα αυτοκίνητα.

Στο κεφάλαιο δύο (2ο) κεφάλαιο γίνεται αναλυτική παρουσίαση του πως είναι στημένη μια υποδομή και από ποιες τεχνολογίες επηρεάζονται οι τεχνολογίες εικονικοποίησης και οι τεχνολογίες αναγνώρισης αντικειμένων. Την αρχιτεκτονική NFV, το Open Source MANO, τι είναι το Edge Computing και ποια τα πλεονεκτήματα του. Τι είναι το Open Stack και τέλος γίνεται αναφορά στις πλατφόρμες εφαρμογής αναγνώρισης αντικειμένων Tensorflow και OpenCv.

Στο κεφάλαιο τρία (3ο) θα γίνει αναφορά για το περιβάλλον του COSMOS, μία υπηρεσία αναγνώρισης αντικειμένων , ένα έργο που υλοποιήθηκε βασισμένο σε όλες αυτές τις τεχνολογίες και παρουσιάστηκαν τον τρόπο λειτουργίας του. Το έργο του COSMOS στόχευε στο σχεδιασμό και τη δοκιμή μιας εφαρμογής με δυνατότητα IoT για έξυπνες τουριστικές περιοχές αξιοποιώντας τις εγκαταστάσεις και τις λειτουργίες NFV / SDN του 5GINFIRE. Την εκτίμηση κινητικότητας χρηστών, την απόφαση εκφόρτωσης και την πρόβλεψη φόρτου και την αρχιτεκτονική του.

Στο (4ο) τέταρτο κεφάλαιο θα γίνει ανάπτυξη της υλοποίησης και εγκατάστασης της υπηρεσίας σε φορητή συσκευή για τις ανάγκες του πειράματος που πραγματεύεται αυτή η εργασία. Χρησιμοποιώντας την πλατφόρμα Tensorflow , η εκπαίδευση του μοντέλου, ο σχολιασμός των εικόνων (Annonation) και πως το μοντέλο αποθηκεύεται και γίνεται served στον εξυπηρετητή που τρέχει την υπηρεσία αναγνώρισης εικόνας.

Στο κεφάλαιο πέντε (5ο) εκτελούνται αποστολές requests στον εξυπηρετητή που εκτελεί την αναγνώριση υπηρεσίας εικόνας, αποστέλλονται ομάδες εικόνων και λαμβάνεται ο μέσος όρος των αποτελεσμάτων σε διάθεση πόρων επεξεργαστικής ισχύ CPU (Ghz), μνήμης (MB) και σε απόκριση χρόνου αναγνώρισης εικόνας. Στο ίδιο κεφάλαιο γίνεται καταγραφή των μετρήσεων και η παρουσίαση τους σε γραφήματα.

Στο κεφάλαιο έξι (6ο) παρουσιάζονται μελλοντικές ενέργειες για την εξέλιξη του πειράματος με συγκεκριμένα εργαλεία για την αξιολόγηση κατανάλωσης πόρων τα οποία βρίσκονται σε ανάπτυξη. Γίνεται αναφορά βάση πρόσφατης έρευνας η εξέλιξη των παραπάνω τεχνολογιών πώς επηρεάζουν την παγκόσμια αγορά και οικονομία. Τέλος, πώς η παγκόσμια πανδημία covid-19 πως επισπεύδει την ανάπτυξη υπηρεσιών που χρησιμοποιούν τεχνολογίες για την κάλυψη απομακρυσμένων υπηρεσιών σε όλους τους τομείς υγείας, οικονομίας, κοινωνίας.

Τέλος, στο κεφάλαιο επτά (7ο) δίνεται όλη η βιβλιογραφία και οι πηγές που χρησιμοποιήθηκαν για την ολοκλήρωση της παρούσας διατριβής.

2 Υποδομή -Θεωρητικό Υπόβαθρο

2.1 Τεχνολογίες Εικονικοποίησης -Ιστορικά Στοιχεία

Τον Οκτώβριο του 2012, η ομάδα προδιαγραφών "Network Function Virtualization"[5], δημοσίευσε μια λευκή βίβλο σε ένα συνέδριο στο Ντάρμστατ της Γερμανία σχετικά με τη δικτύωση ορισμένη με λογισμικό (SND) και το πρωτόκολλο OpenFlow. Η ομάδα αυτή, μέλος του Ευρωπαϊκού Τηλεπικοινωνιακού Ινστιτούτου Προτύπων (ETSI[6])[50], αποτελείται από εκπροσώπους οργανισμών τηλεπικοινωνιών από την Ευρώπη και άλλα μέρη του κόσμου.

Από τη δημοσίευση της λευκής βίβλου, η ομάδα έχει δημιουργήσει αρκετό υλικό με μεγαλύτερες λεπτομέρειες συμπεριλαμβανομένου και ενός πρότυπου ορισμού ορολογίας καθώς και περιπτώσεις χρήσεων που αποτελούν αναφορά για κατασκευαστές και παρόχους υπηρεσιών

Η δύναμη του NFV είναι ότι επιτρέπει την πλήρη αυτοματοποίηση πολλών διαδικασιών που ήταν προηγουμένως χειροκίνητα και αργά. Αυτές οι διαδικασίες περιλαμβάνουν την ανάπτυξη ειδικών λειτουργιών στη λειτουργικότητα το οποίο αποτελεί τη βάση πολλών από τις πολλές αναμενόμενες εφαρμογές του Διαδικτύου. Σε αντίθεση με τις τρέχουσες μη αυτόματες διαδικασίες, την ανάπτυξη ειδικής λειτουργικής λειτουργίας όπου και αν βρίσκεστε κατάλληλη στο δίκτυο υπό NFV μπορεί να είναι φθηνή και γρήγορη χρησιμοποιώντας πλήρη αυτοματοποίηση. Επομένως, είναι NFV η οποία αποτελεί βασική συνιστώσα της τεχνολογίας την αναμενόμενη έκρηξη στις υπηρεσίες με 5G. Φυσικά, το NFV δεν περιορίζεται στην πρόσβαση μέσω κινητού τηλεφώνου και αυτή η αυτοματοποίηση της διαδικασίας θα έχει παρόμοιο αντίκτυπο στα σταθερά δίκτυα .

2.2 Αρχιτεκτονική NFV

Στον κλάδο των τηλεπικοινωνιών η Εικονικοποίηση δικτυακών λειτουργιών (Network Function Virtualization, NFV7) είναι αρχιτεκτονική δικτύου που χρησιμοποιεί τις τεχνολογίες εικονικοποίησης για να εξομοιώσει λειτουργίες κόμβων δικτύων σε δομικά στοιχεία που μπορούν να συνδεθούν μαζί για να δημιουργήσουν υπηρεσίες τηλεπικοινωνιών.

Η εικονικοποίηση των λειτουργιών του δικτύου (NFV) έχει ως στόχο την υλοποίηση των λειτουργιών του δικτύου στο λογισμικό και διευκολύνει το μεγάλο όγκο κυκλοφορίας των τυπικών διακομιστών ενός δικτύου. Αποτελεί μια δικτυακή αρχιτεκτονική που χρησιμοποιεί την

[5] https://en.wikipedia.org/wiki/Network_function_virtualization

[6] https://www.etsi.org/

[7] https://www.etsi.org/deliver/etsi_gs/NFV/001_099/001/01.01.01_60/gs_NFV001v010101p.pdf

τεχνολογία της εικονικοποίησης της Τεχνολογίας της Πληροφορίας (IT) προκειμένου να εικονικοποιήσει ολόκληρες κατηγορίες των λειτουργιών κόμβου του δικτύου σε δομικά στοιχεία που μπορούν να συνδεθούν ή να λειτουργήσουν σε αλυσίδα για να δημιουργήσουν υπηρεσίες επικοινωνιών. Η τεχνολογία NFV βασίζεται, αλλά ταυτόχρονα διαφέρει από παραδοσιακές τεχνικές οπτικοποίησης του διακομιστή (server-virtualization) και μπορεί να αποτελείται από μια ή περισσότερες εικονικές μηχανές που εκτελούν διαφορετικό λογισμικό και διεργασίες πάνω σε τυπικούς διακομιστές υψηλής χωρητικότητας ή ακόμη και σε υποδομές υπολογιστικού νέφους, αντί να διαθέτουν προσαρμοζόμενες συσκευές υλικού για κάθε λειτουργία του δικτύου.

Η NFV βασίζεται σε παραδοσιακές τεχνικές εικονικοποίησης υπολογιστών, όπως αυτές χρησιμοποιούνται στην πληροφορική, αλλά εμφανίζει και διαφορές. Μια εικονική λειτουργία δικτύου, ή VNF, μπορεί να αποτελείται από μία ή περισσότερες εικονικές μηχανές που τρέχουν διαφορετικό λογισμικό και διεργασίες, πάνω σε κοινούς εξυπηρετητές, μεταγωγείς και συσκευές αποθήκευσης, ή ακόμα και σε υπολογιστικά νέφη, αντί να χρειάζεται εξειδικευμένες συσκευές για κάθε λειτουργία του δικτύου.

Παραδείγματα δικτυακών λειτουργιών που εικονικοποιούνται είναι εξισορροπητές φόρτου, τείχη προστασίας, συστήματα ανίχνευσης εισβολής και επιταχυντές διαδικτύου.

Η ανάπτυξη προϊόντων στον κλάδο των τηλεπικοινωνιών παραδοσιακά ακολουθούσε αυστηρά πρότυπα σταθερότητας, συμβατότητας με πρωτόκολλα και ποιότητας. Παρ' όλο που αυτό το μοντέλο κατασκευής λειτούργησε καλά στο παρελθόν, αναμφίβολα οδηγούσε σε μεγάλους κύκλους ανάπτυξης, αργούς ρυθμούς προόδου και εξάρτηση από ιδιόκτητο ή εξειδικευμένο υλικό, π.χ. ολοκληρωμένα κυκλώματα. Ο ανταγωνισμός στο χώρο των επικοινωνιών από ευέλικτους οργανισμούς που δραστηριοποιούνται σε μεγάλη κλίμακα στο διαδίκτυο (όπως το Google Talk, το Skype, το Netflix) ανάγκασε τους παρόχους υπηρεσιών να αναζητήσουν τρόπους να αλλάξουν αυτή την κατάσταση .

2.2.1 Πλαίσιο Λειτουργίας NFV

Το πλαίσιο λειτουργίας της NFV αποτελείται από 3 βασικά μέρη:

- Τις εικονικοποιημένες δικτυακές λειτουργίες (VNFs) οι οποίες είναι υλοποιήσεις σε λογισμικό διάφορων δικτυακών λειτουργιών που μπορούν να αναπτυχθούν πάνω σε υποδομή εικονικοποίησης δικτυακών λειτουργιών (NFVI).
- Την υποδομή εικονικοποίησης δικτυακών λειτουργιών που είναι το σύνολο του υλικού και λογισμικού που συνθέτει το περιβάλλον όπου αναπτύσσονται οι VNFs. Η υποδομή NFV μπορεί να εκτείνεται σε περισσότερες από μια τοποθεσίες ενώ το δίκτυο που παρέχει συνδεσιμότητα ανάμεσα σε αυτές τις τοποθεσίες θεωρείται μέρος της υποδομής.

- Το πλαίσιο διαχείρισης και ενορχήστρωσης της εικονικοποίησης δικτυακών λειτουργιών (NFV-MANO) που είναι το σύνολο όλων των λειτουργικών μονάδων, των δεδομένων που αυτά χρησιμοποιούν, των σημείων αναφοράς και των διεπαφών μέσω των οποίων αυτές ανταλλάσσουν πληροφορίες με σκοπό τη διαχείριση και την ενορχήστρωση των VNFs και της υποδομής NFV[51].

Η βασική μονάδα τόσο για την υποδομή NFV όσο και για το NFV-MANO είναι η πλατφόρμα NFV. Από την πλευρά της υποδομής NFV, αυτή αποτελείται από τους εικονικούς και φυσικούς πόρους επεξεργασίας και αποθήκευσης και από λογισμικό εικονικοποίησης. Από την πλευρά του NFV-MANO, η πλατφόρμα NFV αποτελείται από το λογισμικό διαχείρισης και εικονικοποίησης. Η πλατφόρμα NFV υλοποιεί λειτουργίες που την καθιστούν κατάλληλη για χρήση σε τηλεπικοινωνιακό περιβάλλον, όπως διαχείριση και επίβλεψη των διάφορων συνθετικών της πλατφόρμας, ανάνηψη από σφάλματα και αποτελεσματική ασφάλεια, στοιχεία απαραίτητα για ένα δίκτυο δημόσιας χρήσης.

Αν γίνει μία πιο πρακτική προσέγγιση μπορεί να ειπωθεί ότι ένας πάροχος υπηρεσιών που ακολουθεί την NFV αρχιτεκτονική υλοποιεί μια ή περισσότερες εικονικές δικτυακές λειτουργίες, ή VNFs. Μια VNF από μόνη της δεν προσφέρει υποχρεωτικά κάποιο προϊόν ή υπηρεσία στους πελάτες του παρόχου. Για να δημιουργηθούν πιο περίπλοκες υπηρεσίες, χρησιμοποιείται η έννοια της αλυσίδας υπηρεσιών, όπου πολλαπλά VNFs χρησιμοποιούνται στη σειρά για να δώσουν μια υπηρεσία.

Μια άλλη πτυχή της εφαρμογής της NFV είναι η ενορχήστρωση. Για την κατασκευή αξιόπιστων και ολοκληρωμένων υπηρεσιών, η NFV απαιτεί ότι το δίκτυο είναι σε θέση να ξεκινήσει VNFs, να τις παρακολουθεί, να τις επισκευάζει, και (το πιο σημαντικό για έναν πάροχο υπηρεσιών) να χρεώνει για τις υπηρεσίες που παρέχονται. Αυτά τα χαρακτηριστικά, που αναφέρονται ως "επιπέδου-παρόχου" (carrier-grade) χαρακτηριστικά, ανατίθενται σε ένα επίπεδο ενορχήστρωση, ώστε να παρέχει υψηλή διαθεσιμότητα και ασφάλεια, και χαμηλό κόστος λειτουργίας και συντήρησης. Το σημαντικότερο, η ενορχήστρωση του στρώματος πρέπει να είναι σε θέση να διαχειριστεί VNFs ανεξάρτητα από την υποκείμενη τεχνολογία που υλοποιεί η VNF. Για παράδειγμα, ένα σύστημα ενορχήστρωσης πρέπει να είναι σε θέση να διαχειριστεί ένα Firewall VNF από τον κατασκευαστή Χ που τρέχει σε VMware[8] vSphere εξίσου καλά όπως και ένα IMS VNF από τον κατασκευαστή Ψ το οποίο τρέχει στο KVM[52].

[8] https://el.wikipedia.org/wiki/VMware

2.2.2 Κατανεμημένη NFV

Η αρχική αντίληψη για την NFV ήταν ότι η ικανότητα εικονικοποίησης θα πρέπει να εφαρμόζεται σε κέντρα δεδομένων. Αυτή η προσέγγιση λειτουργεί σε πολλές, αλλά όχι σε όλες τις περιπτώσεις. Η NFV προϋποθέτει και τονίζει την ευρύτερη δυνατή ευελιξία ως προς τη φυσική θέση του εικονικοποιημένων λειτουργιών.

Ιδανικά, ως εκ τούτου, εικονικοποιημένες λειτουργίες θα πρέπει να βρίσκονται στο σημείο που είναι πιο αποτελεσματικές και λιγότερο δαπανηρές. Αυτό σημαίνει ότι ένας πάροχος υπηρεσιών θα πρέπει να είναι ελεύθερος να αναπτύξει NFV σε όλες τις πιθανές τοποθεσίες, από το κέντρο δεδομένων, στους κόμβους του δικτύου έως και στις εγκαταστάσεις του καταναλωτή. Η προσέγγιση αυτή, που είναι γνωστή ως κατανεμημένη NFV, έχει τονιστεί από την αρχή καθώς η NFV αναπτυσσόταν, και κατέχει εξέχουσα θέση στα τελευταία έγγραφα της ομάδας NFV.

Για ορισμένες περιπτώσεις, υπάρχουν σαφή πλεονεκτήματα για ένα πάροχο υπηρεσιών αν αναπτύξει εικονικοποιημένες λειτουργίες στις εγκαταστάσεις του πελάτη. Τα πλεονεκτήματα αυτά κυμαίνονται από οικονομικά έως και την απόδοση.

2.2.3 Οφέλη Σπονδυλωτής NFV

Κατά το σχεδιασμό και την ανάπτυξη του λογισμικού που παρέχουν οι VNFs, οι κατασκευαστές μπορούν να δομήσουν το λογισμικό σε υποσυστήματα και τα υποσυστήματα αυτά να τα παρέχουν εικονικές μηχανές. Αυτά τα υποσυστήματα ονομάζονται VNF Υποσυστήματα (VNFCs). Μια VNF υλοποιείται με ένα ή περισσότερα VNFCs.

Τα VNFCs θα πρέπει γενικά να είναι σε θέση να κλιμακοθούν οριζόντιως και καθέτως. Με το να είναι σε θέση να διαθέσει εύκαμπτο (virtual) Επεξεργαστές σε κάθε VNFC περιπτώσεις, η διαχείριση του δικτύου στρώμα μπορεί κλίμακας (δηλαδή, την κλίμακα κάθετα) το VNFC για να παρέχει την απόδοση/την απόδοση και επεκτασιμότητα προσδοκίες για ένα ενιαίο σύστημα ή μια ενιαία πλατφόρμα. Ομοίως, η διαχείριση του δικτύου στρώμα μπορεί κλίμακας (δηλαδή, κλίμακα οριζόντια) VNFC ενεργοποιώντας πολλές περιπτώσεις τέτοιων VNFC σε πολλαπλές πλατφόρμες και ως εκ τούτου να φτάσει τις επιδόσεις και την αρχιτεκτονική προδιαγραφές, ενώ δεν θέτει σε κίνδυνο την άλλη VNFC λειτουργία σταθερότητες.

2.2.4 Εφαρμογές NFV στον κλάδο των επικοινωνιών

Η NFV έχει αποδειχθεί δημοφιλές πρότυπο από πολύ νωρίς. Οι άμεσες εφαρμογές της είναι πολλές, όπως οι εικονικοποίηση των σταθμών βάσης κινητής τηλεφωνίας, των πλατφόρμων ως υπηρεσία (PaaS), των δικτύων διανομής περιεχομένου (CDN), της σταθερής τηλεφωνίας και του οικιακού εξοπλισμού. Τα πιθανά οφέλη της NFV αναμένεται να είναι σημαντικά. Η εικονικοποίηση των λειτουργιών δικτύου πάνω σε υλικό γενικού σκοπού αναμένεται να μειώσει τα έξοδα κτίσης και τις λειτουργικές δαπάνες, καθώς και το χρόνο δημιουργίας υπηρεσιών και προϊόντων. Πολλοί σημαντικοί κατασκευαστές εξοπλισμού δικτύου έχουν ανακοινώσει την υποστήριξη της NFV. Αυτό συνέπεσε με ανακοινώσεις σχετικά με την NFV από μεγάλους προμηθευτές λογισμικού που παρέχουν τις NFV πλατφόρμες που χρησιμοποιούνται από τους προμηθευτές εξοπλισμού για την κατασκευή των δικών τους NFV προϊόντων.

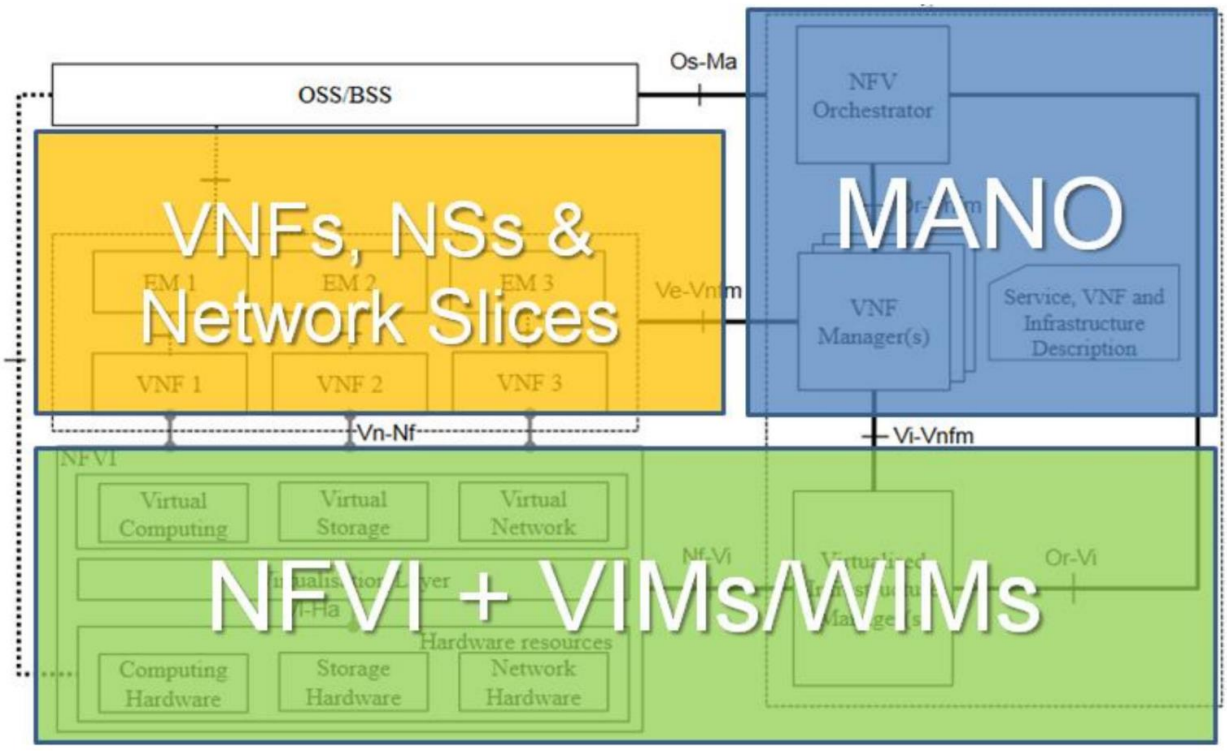
Ωστόσο, για να υλοποιηθούν τα προσδοκώμενα οφέλη της εικονικοποίησης, οι κατασκευαστές δικτυακού εξοπλισμού βελτιώνουν τις τεχνολογίες εικονικοποίησης για να ενσωματώσουν τα χαρακτηριστικά "επιπέδου-παρόχου" που απαιτούνται για να επιτευχθεί η υψηλή διαθεσιμότητα, η επεκτασιμότητα, η απόδοση και η αποτελεσματική διαχείριση του δικτύου. Για να ελαχιστοποιηθεί το συνολικό κόστος ιδιοκτησίας (TCO), τα χαρακτηριστικά "επιπέδου-παρόχου" πρέπει να υλοποιηθούν όσο το δυνατόν πιο αποτελεσματικά. Αυτό απαιτεί από τις NFV λύσεις να κάνουν αποδοτική χρήση των πλεοναζόντων πόρων για την επίτευξη διαθεσιμότητας 99.999%, χωρίς συμβιβασμούς στις συνεπείς επιδόσεις.

Η NFV πλατφόρμα είναι η βάση για την επίτευξη αποτελεσματικών NFV λύσεων "επιπέδου-παρόχου". Είναι μια πλατφόρμα λογισμικού που εκτελείται σε τυπικό υλικό πολλαπλών πυρήνων και κατασκευάστηκε με τη χρήση λογισμικού ανοιχτού κώδικα που ενσωματώνει χαρακτηριστικά "επιπέδου-παρόχου". Η NFV πλατφόρμα είναι υπεύθυνη για τη δυναμική ανακατανομή VNFs λόγω σφαλμάτων και αλλαγών στην κίνηση του δικτύου, και ως εκ τούτου, διαδραματίζει σημαντικό ρόλο στην επίτευξη υψηλής διαθεσιμότητας. Υπάρχουν πολλές πρωτοβουλίες σε εξέλιξη για να καθορίσουν, να ευθυγραμμίσουν και να προωθήσουν NFV δυνατότητες "επιπέδου-παρόχου" όπως τα ETSI NFV Proof of Concept, ATIS Open Platform for NFV,το Carrier Network Virtualization Awards και διάφορα οικοσυστήματα κατασκευαστών.

Ο εικονικός μεταγωγέας (vSwitch), ένα βασικό συστατικό των NFV πλατφορμών, είναι υπεύθυνος για την παροχή συνδεσιμότητας μεταξύ των εικονικών μηχανών αλλά και μεταξύ των εικονικών μηχανών και του εξωτερικό δίκτυο. Η απόδοση του καθορίζει τόσο το εύρος ζώνης των VNFs όσο και το κόστος και την αποτελεσματικότητα των NFV λύσεων. Η τυπική απόδοση του Open vSwitchs (OVS) έχει αδυναμίες που πρέπει να επιλυθούν για να καλύψει τις ανάγκες

της υποδομής NFV. Σημαντικές βελτιώσεις στην απόδοση έχουν αναφερθεί από κατασκευαστές NFV τόσο για το OVS όσο και για την επιταχυνόμενη Open vSwitch (AVS) έκδοση.

Επίσης η εικονικοποίηση αλλάζει τον τρόπο που η διαθεσιμότητα καθορίζεται, μετράτε και επιτυγχάνεται σε NFV λύσεις. Καθώς VNFs αντικαθιστούν τις παραδοσιακές λειτουργίες, υπάρχει μια μετατόπιση από διαθεσιμότητα που βασίζεται σε πλεονάζοντα εξοπλισμό σε διαθεσιμότητα βασισμένη σε πλεονάζουσες υπηρεσίες και λογισμικό. Η εικονικοποίηση λειτουργιών δικτύου σπάει τη σύζευξη με ειδικό εξοπλισμό, ως εκ τούτου, η διαθεσιμότητα ορίζεται από τη διαθεσιμότητα των υπηρεσιών που προσφέρουν οι VNFs. Επειδή η NFV τεχνολογία μπορεί να εικονικοποιήσει ένα ευρύ φάσμα λειτουργιών δικτύων, το καθένα με τη δική του απαίτηση διαθεσιμότητας, οι NFV πλατφόρμες θα πρέπει να υποστηρίζουν ένα ευρύ φάσμα επιλογών σε ανοχή ελαττωμάτων. Αυτή η ευελιξία επιτρέπει στους παρόχους τηλεπικοινωνιών για να βελτιστοποιήσουν τις NFV λύσεις για να καλύψουν κάθε απαίτηση διαθεσιμότητας.[52]



Εικόνα 2-1:NFV partitioning[9]

[9] https://osm.etsi.org/images/OSM_EUAG_White_Paper_OSM_Scope_and_Functionality.pdf

2.3 Πλατφόρμα MANO (OSM)

Το Open Source Mano [56]είναι μια πρωτοβουλία που φιλοξενείται από το ETSI για την ανάπτυξη ενός λογισμικού ανοιχτού κώδικα NFV διαχείρισης και ενορχήστρωσης (MANO) ευθυγραμμισμένης με το ETSI NFV.

Δύο από τα βασικά στοιχεία του αρχιτεκτονικού πλαισίου και των προτύπων του ETSI NFV είναι ο NFV Orchestrator και ο VNF Manager, γνωστός ως NFV MANO. Επιπρόσθετα επίπεδα, όπως η ενορχήστρωση των υπηρεσιών, απαιτούνται επίσης για τους φορείς εκμετάλλευσης για την ενεργοποίηση πραγματικών υπηρεσιών NFV. Το λογισμικό ανοιχτού κώδικα μπορεί να διευκολύνει την εφαρμογή μιας αρχιτεκτονικής NFV ευθυγραμμισμένης με το ETSI, να παράσχει πρακτική και ουσιαστική ανατροφοδότηση στο ETSI ISG NFV και να αυξήσει την πιθανότητα δια λειτουργικότητάς μεταξύ των εφαρμογών NFV.

Η ομάδα Open Mano (OSM) του ETSI, αναπτύσσει μια στοίβα διαχείρισης NFV και ενορχήστρωσης ανοιχτού κώδικα χρησιμοποιώντας καλά εγκατεστημένα εργαλεία ανοιχτού κώδικα και διαδικασίες εργασίας. Η δραστηριότητα είναι ευθυγραμμισμένη με την εξέλιξη του ETSI NFV και θα παρέχει μια τακτικά ενημερωμένη εφαρμογή αναφοράς του NFV MANO. Ο OSM στοχεύει στο να επιτρέψει σε ένα οικοσύστημα λύσεων NFV να παράσχουν γρήγορα και οικονομικά αποδοτικές λύσεις στους χρήστες τους.

Αξίζει να το αναφερθεί ότι κυκλοφόρησε στο GitHub (υπηρεσία φιλοξενίας Git-repository hosting) υπό την άδεια Apache 2. Για μια ολοκληρωμένη επισκόπηση των λειτουργιών OSM, θα πρέπει αξιολογηθούν τα white paper του OSM όπου αναλύεται ο σκοπός και λειτουργικότητες ή στις σημειώσεις προηγούμενων εκδόσεων OSM (ONE, TWO, THREE, FOUR, FIVE)[56].

2.3.1 Αλληλεπίδραση με VIM και VNF

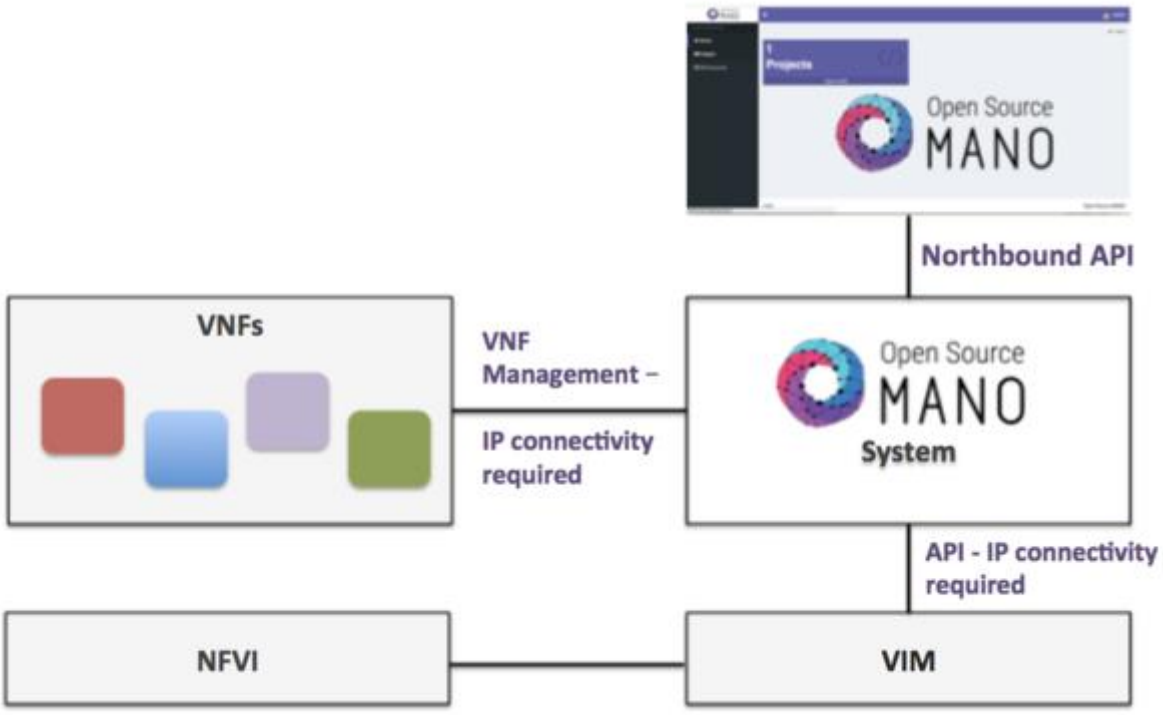
Το OSM μιλάει στο VIM για την ανάπτυξη VNFs και VLs που τα συνδέουν.

Το OSM μιλάει με τα VNF που αναπτύσσονται σε ένα VIM για να τρέξουν τα πρωτόκολλα διαμόρφωσης day-0, day-1 και day-2.

2.3.2 Πλεονεκτήματα OSM

Ο ανοικτός πηγαίος κώδικας και το μοντέλο ανάπτυξης της κοινότητας που όλα τα μέρη βοηθούν στη δοκιμή και στη διόρθωση σφαλμάτων καθιστά το έργο ανοιχτό για όσους θέλουν να εξοικειωθούν.

Όπως προαναφέρθηκε, το λογισμικό OSM είναι μια ευρέως ανοιχτή κοινότητα που κατευθύνεται από παρόχους υπηρεσιών και υποστηρίζονται από βασικούς συντελεστές της βιομηχανίας και του χώρου εικονικοποίησης. Το OSM MANO αγκαλιάζει την πολυπλοκότητα που απαιτείται για τις πεδίο πραγματικό χρόνο και πραγματικά σενάρια. Με υποστήριξη EPA, λειτουργία Multi-VIM, Multi-site της δραστηριότητας και της απόσπασης των RO και SO. Παρέχει επίσης μια διαμόρφωση CLI και ένα GUI εργαλείο, συν είναι multi-vendor καθιστώντας το φιλικό για τους μηχανικούς δικτύων. Επιπλέον, OSMH MANO διαθέτει ένα ολοκληρωμένο σύνολο συνδέσεων δικτύου L2. Αυτός ο συνδυασμός κρύβει χαμηλού επιπέδου πολυπλοκότητα για μηχανικούς δικτύων, ενώ παράλληλα εξασφαλίζει συνεπή ανάπτυξη.



Εικόνα 2-2:Osmtopology[10]

[10] https://osm.etsi.org/wikipub/index.php/OSM_Release_FIVE

Για να λειτουργήσει ο OSM, θεωρείται ότι:

- Κάθε VIM έχει ένα τελικό σημείο API που μπορεί να είναι προσιτό από το OSM.
- Κάθε VIM διαθέτει ένα λεγόμενο δίκτυο διαχείρισης το οποίο παρέχει διεύθυνση IP σε VNFs
- Αυτό το δίκτυο διαχείρισης είναι προσβάσιμο από το OSM

2.4 Υπολογιστική στις Άκρες των Δικτύων

Μία από τις πιο πρόσφατες καινοτομίες είναι το cloud computing, υπολογιστική νέφους. Το Dropbox, το Google Drive και οι υπηρεσίες web της Amazon, εκτελούνται σε περιβάλλον cloud. Με το cloud, δεν αποθηκεύονται οι πληροφορίες στον τοπικό υπολογιστή , αλλά αποθηκεύονται στους κοινόχρηστους διακομιστές που υπάρχουν παγκοσμίως. Το cloud έχει τελικά επικρατήσει, καθώς πολλές μεγάλες εταιρείες και μεμονωμένοι χρήστες έχουν υιοθετήσει αυτή την τεχνολογία.

Παρόλα αυτά, μπορεί να έφτασε ήδη ο καιρός να αντικατασταθεί το cloud με το Internet of Things (IoT), τα δεδομένα πρέπει να αποστέλλονται και να λαμβάνονται πιο γρήγορα από ποτέ. Το Internet of Things μοιάζει με τα συνήθη αντικείμενα της καθημερινότητας, όπως με το ψυγείο, την τοστιέρα, το ρολόι και το αυτοκίνητό, δίνοντας σύνδεση στο internet. Μπορούν για παράδειγμα οι χρήστες να αντιληφθούν γιατί χρειάζεται μια γρήγορη σύνδεση το αυτοκίνητο.

Οι ειδικοί λένε ότι το cloud σιγά σιγά αντικαθίσταται από την επόμενη τεχνολογία Edge Computing [57]

2.4.1 Τι είναι η Υπολογιστική στις Άκρες των Δικτύων

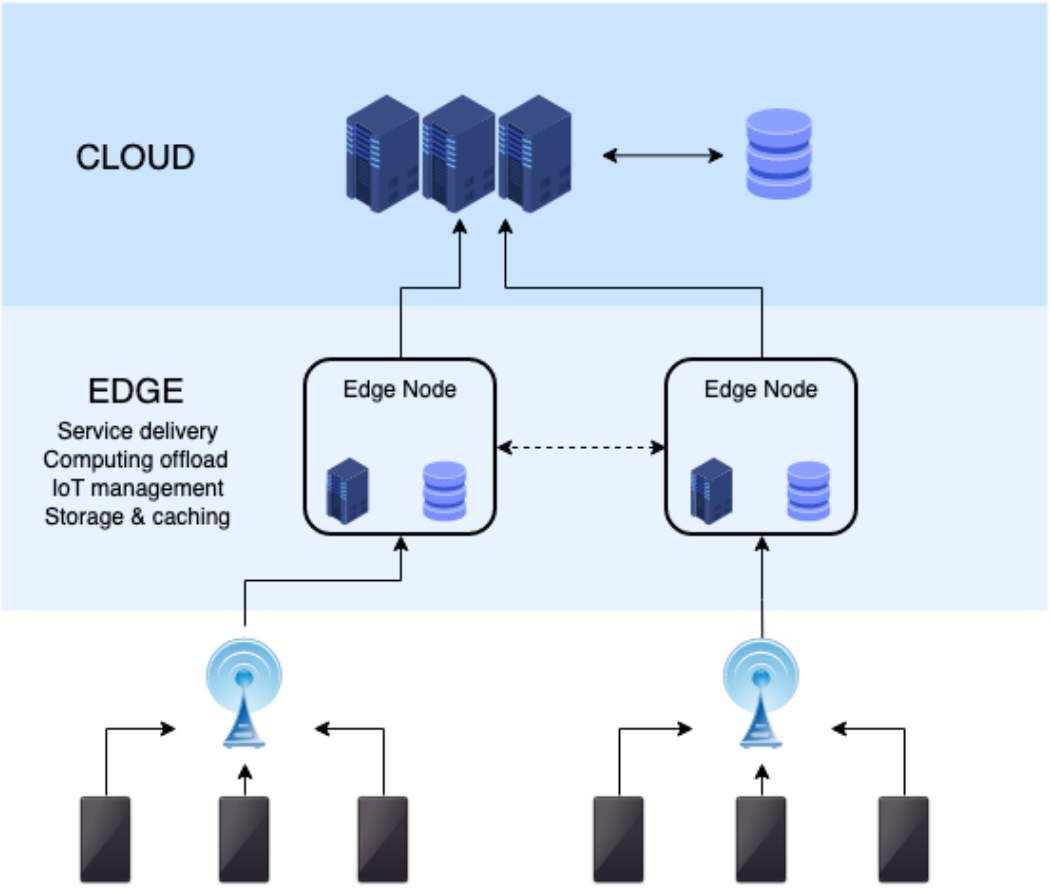
Ο ορισμός του Edge Computing[11] είναι, “ένας κατανεμητής ανοιχτής αρχιτεκτονικής IT που διαθέτει αποκεντρωμένη επεξεργαστική ισχύ, υποστηρίζοντας τις τεχνολογίες του mobile computing και του Internet of Things (IoT)”. Στο Edge Computing, τα δεδομένα υποβάλλονται σε επεξεργασία από την ίδια τη συσκευή ή από έναν τοπικό υπολογιστή ή διακομιστή, αντί να μεταδίδονται σε ένα κέντρο δεδομένων».

Με το Edge Computing, τα δεδομένα που παράγονται από συσκευές IoT, επεξεργάζονται πιο κοντά εκεί που δημιουργούνται, αντί να αποστέλλονται σε μεγάλες διαδρομές στα κέντρα δεδομένων ή στο cloud. Όταν η επεξεργασία γίνεται πιο κοντά στην «άκρη» του δικτύου, αυτό

[11] https://www.cloudwards.net/what-is-edge-computing/

επιτρέπει στις εταιρίες να αναλύσουν τα σημαντικά δεδομένα σε σχεδόν πραγματικό χρόνο, και επιτρέπει στις συσκευές IoT την αποστολή και λήψη δεδομένων πολύ πιο γρήγορα και πιο αποτελεσματικά.

Το Edge Computing διαλέγει τα δεδομένα σε τοπικό επίπεδο, ώστε κάποια από αυτά να επεξεργάζονται σε τοπικό επίπεδο, μειώνοντας έτσι την κυκλοφορία backhaul στο κεντρικό αποθηκευτικό χώρο. Αυτό είναι σημαντικό, διότι μειώνει την εξάρτηση από το Wi-Fi ή τα δεδομένα δικτύου. Δεν θα ήταν αποδεκτό για παράδειγμα να χάνονται οι υπηρεσίες πλοήγησης του αυτοκινήτου σας, επειδή το σήμα είναι ασθενές[58].



Εικόνα 2-3:Edge computing infrastructure[12]

2.4.2 Έννοιες και ορισμοί

Πριν προχωρήσουμε περαιτέρω, ας βεβαιωθούμε ότι γίνονται κατανοητά συγκεκριμένες έννοιες και ορισμοί.

[12] https://en.wikipedia.org/wiki/Edge_computing#/media/File:Edge_computing_infrastructure.png

Συσκευές Edge: Κάθε συσκευή που παράγει τα δεδομένα. Οι συσκευές αυτές θα μπορούσαν να είναι αισθητήρες, βιομηχανικά μηχανήματα, αυτοκίνητα, ρολόγια, ή οποιαδήποτε άλλη συσκευή που παράγει ή συλλέγει δεδομένα.

Edge: Αυτό εξαρτάται από την κάθε περίπτωση. Στις επικοινωνίες, το Edge μπορεί να είναι ένα κινητό τηλέφωνο ή ένα cell tower. Στην αυτοκινητοβιομηχανία, το Edge μπορεί να είναι ένα αυτοκίνητό. Στη βιομηχανία, θα μπορούσε να είναι ένα μηχάνημα κάποιας επιχείρησης. Στον τομέα της πληροφορικής, το Edge μπορεί να είναι ένα laptop.

Edge Gateway: Αυτό είναι το buffer μεταξύ του σημείου όπου γίνεται η επεξεργασία του Edge και το ευρύτερου δικτύου cloud.

Δίκτυο cloud: Το δίκτυο cloud αφορά στις συνδέσεις δικτύου μεταξύ των συσκευών Edge και το cloud.

Fat client: Είναι το λογισμικό που μπορεί να κάνει την επεξεργασία δεδομένων στις Edge συσκευές. Ένας λεπτός client μεταφέρει μόνο τα δεδομένα.

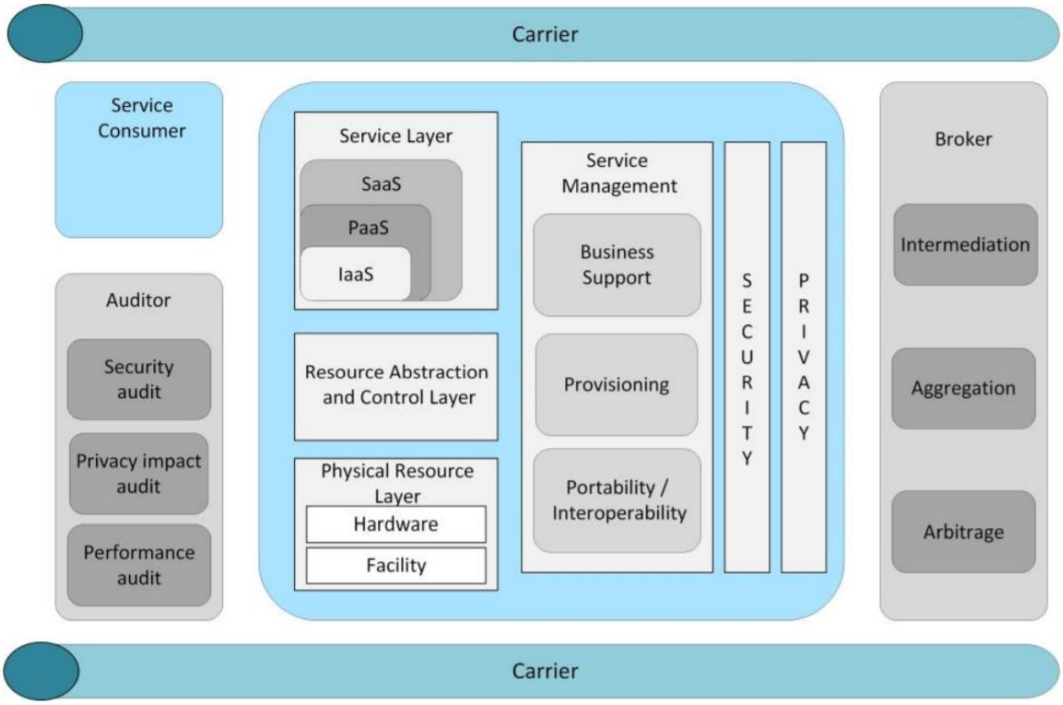
Εξοπλισμός Edge: Οι περισσότερες συσκευές και μηχανήματα μπορούν να λειτουργήσουν σε ένα υπολογιστικό περιβάλλον τύπου Edge, καθιστώντας τα προσβάσιμα στο internet. Συστήματα υπολογιστών, όπως η υπηρεσία "Amazon Web Service's Snowball", μπορεί να χρησιμοποιηθεί σε Edge Computing διατάξεις. Πολλά πράγματα στο σπίτι, συμπεριλαμβανομένου του κουδουνιού της πόρτας, της τοστιέρας, των ηλεκτρικών συσκευών και του αυτοκινήτου στο γκαράζ, μπορεί να διαθέτουν εξοπλισμό "edge" τεχνολογίας.

Mobile Edge Computing: Αυτή είναι η προέκταση του Edge Computing σε δίκτυα 5G [59].

2.4.3 Πλεονεκτήματα της Υπολογιστικής των Άκρων

Αρκετές τεχνολογίες, όπως συστάδας (cluster), πλέγματος (grid) και σήμερα υπολογιστικής νέφους (cloud computing), επιτρέπουν την πρόσβαση σε τεράστια υπολογιστική ισχύ με τη χρήση της εικονικοποίησης, με τη συσσώρευση πόρων και την οπτική μεμονωμένου συστήματος. Με την υιοθέτηση της υπολογιστικής κοινής ωφέλειας (utility computing), η κατόπιν αιτήσεως (on-demand) προσφορά υπολογιστικής ισχύος είναι δυνατή, με τους χρήστες να χρεώνονται με βάση τη χρήση (paying-as-you-go). Η υπολογιστική νέφους χρησιμοποιείται ευρέως ως όρος με το κυρίως μοντέλο να προσφέρει δυνατότητες υπολογισμών, αποθήκευσης δεδομένων, και λογισμικού 'σαν υπηρεσία' ('as a service') στον χρήστη. Το National Institute of Standards and Technology (NIST) χαρακτηρίζει το νέφος ως εξής 'μοντέλο με πληρωμή ανάλογα της χρήσης για την ενεργοποίηση, βολικής, κατ' απαίτηση πρόσβασης μέσω δικτύου σε δεξαμενή διαμοιρασμένων υπολογιστικών πόρων (π.χ. δικτύων, εξυπηρετητών, αποθηκευτικού χώρου, εφαρμογών, υπηρεσιών), που μπορούν γρήγορα να διατεθούν και να αποδεσμευθούν με

ελάχιστη διαχειριστική προσπάθεια ή την επέμβαση του παρόχου της υπηρεσίας. Στην Εικόνα 2.4 αποτυπώνεται το μοντέλο αναφοράς του οργανισμού NIST[61].



Εικόνα 2-4:. NIST μοντέλο αναφοράς[13]

Οι ρίζες της υπολογιστικής νέφους ακολουθούν τις εξελίξεις προς την επόμενη γενιά του διαδικτύου με έμφαση το περιεχόμενο (content-centric future internet), συγκεκριμένα:

Μετατόπιση από μεγάλα υπολογιστικά συστήματα σε νέφη

Η πληροφορική υποστηρίζει την κίνηση από υπολογιστική στο εσωτερικό ενός οργανισμού στην παροχή υπηρεσιών κοινής ωφέλειας με χρήση του διαδικτύου.

SOA, Web Services, Web 2.0, και Mashups

Η εξέλιξη των υπηρεσιών διαδικτύου χρησιμοποιείται για την υποστήριξη εφαρμογών που τρέχουν σε διαφορετικές πλατφόρμες, ενώ η ωρίμανση σχετικών προτύπων επιτρέπει τη δημιουργία δυναμικών υπηρεσιών, οι οποίες επιτρέπουν τόσο την πρόσβαση σε πραγματικό χρόνο με βάση σχετική αίτηση όσο και σύνθεση πολλαπλών νέων υπηρεσιών.

Υπολογιστική πλέγματος (Grid computing)

Η υπολογιστική πλέγματος επενδύει στη συγκέντρωση κατανεμημένων πόρων και τη διάφανη χρήση τους. Η ανάπτυξη προτύπων και τεχνολογιών εικονικοποίησης οδήγησε στην ωρίμανση της προσέγγισης σε ποικίλα θέματα, για παράδειγμα, όπως τη δια λειτουργικότητα.

[13] *The Cloud Computing Conceptual Reference Model, https://www.kurzweilai.net/nist-issues-government-cloud-computing-roadmap-and-architecture*

Κοινωφελής υπολογιστική (Utility computing)

Η συγκεκριμένη προσέγγιση βελτιστοποιεί την υπολογιστική πλέγματος βασισμένη στην αξία κάθε προγραμματισμένης εργασίας και το αντίστοιχο κόστος για τον χρήστη.

Εικονικοποίηση υλικού (hardware virtualization)

Η υπολογιστική νέφους υποστηρίζεται από μεγάλα κέντρα δεδομένων (data centers) που εξυπηρετούν πολλούς χρήστες και φιλοξενούν χιλιάδες εφαρμογές. Άλλωστε, γι' αυτό τον λόγο η εικονικοποίηση υλικού επιτρέπει τον διαχωρισμό πόρων και την ανάθεσή τους σε χρήστες και εφαρμογές (πχ. λειτουργία πολλαπλών λειτουργικών συστημάτων σε ένα φυσικό σύστημα). Κατά την υλοποίηση αφιερώνεται ειδικό στρώμα λογισμικού, γνωστό και ως διαχειριστής εικονικής μηχανής (Virtual Machine Monitor – VMM - ή hypervisor),που διαχειρίζεται την πρόσβαση στους φυσικούς πόρους, ώστε κάθε φιλοξενούμενο λειτουργικό σύστημα να βλέπει μια εικονική μηχανή (Virtual Machine), ένα σύνολο από διεπαφές της εικονικής πλατφόρμας. Ενδεικτικά γνωστές πλατφόρμες είναι η VMWare, Xen και KVM.

Εικονικές συσκευές και ανοικτό περιβάλλον εικονικοποίησης

Σύγχρονες αγορές παρέχουν με τη μορφή εφαρμογών εικονικών συσκευών (virtual appliances applications) συνδυασμένες με τους απαραίτητους πόρους, μορφοποιημένους σαν μια εικόνα δίσκου (VM disk image) για να αναπτυχθεί στον διαχειριστή της εικονικής μηχανής. Προς αυτή την κατεύθυνση παρέχονται έτοιμα πακέτα από τον Open Virtualization Format (OVF) με στοιχεία, όπως απαραίτητα χαρακτηριστικά υλικού, λειτουργικό σύστημα, διαδικασία εκκίνησης και λήξης, εικονικούς δίσκους μαζί με τις σχετικές άδειες χρήσης και τους οδηγούς λειτουργίας.

Αυτόνομη υπολογιστική (Autonomic computing)

Στόχος της τεχνολογίας είναι η ελαχιστοποίηση της ανθρώπινης παρέμβασης στη λειτουργία υπολογιστικού συστήματος, χρησιμοποιώντας τεχνικές εποπτείας και προσαρμογής με στόχο τη βελτιστοποίηση της απόδοσης του συστήματος.

Με τη νέα τεχνολογία, έρχονται νέα πλεονεκτήματα. Παρακάτω αναλύονται 5 πλεονεκτήματα του Edge Computing που ξεχωρίζουν.

Χρόνος απόκρισης: Χωρίς αμφίδρομο cloud, το Edge Computing μειώνει τον λανθάνοντα χρόνο και παράγει πιο γρήγορα αποκρίσεις. Αυτό θα βοηθήσει στη συνέχεια λειτουργίας μετά από διακοπές που μπορούν να συμβούν ή από επικίνδυνα περιστατικά που λαμβάνουν χώρα.

Αξιόπιστες λειτουργίες με περιορισμένη σύνδεση: Αναξιόπιστη σύνδεση στο internet σε δυσπρόσιτες περιοχές όπως πετρελαιοπηγές, αντλίες αγροκτημάτων, ηλιακά πάρκα ή ανεμόμυλους, κάνει τα πράγματα δύσκολα. Η δυνατότητα αποθήκευσης και επεξεργασίας δεδομένων στο "όριο" εξασφαλίζει ότι δεν υπάρχει απώλεια δεδομένων ή λειτουργική βλάβη σε περίπτωση που υπάρχει περιορισμένη σύνδεση στο internet.

Ασφάλεια: Το Edge Computing, μπορεί να αποτρέψει την μεταφορά δεδομένων μεταξύ των συσκευών και του cloud. Μπορείτε να φιλτράρετε ευαίσθητες πληροφορίες σε τοπικό επίπεδο, και να επιλέξετε ποιες πληροφορίες θα μεταφέρονται στο cloud. Αυτό θα επιτρέψει στους χρήστες να δημιουργήσουν το κατάλληλο πλαίσιο ασφάλειας, απαραίτητο για την ασφάλεια και τους ελέγχους για παράδειγμα στις επιχειρήσεις.

Αποτελεσματικό Κόστος: Μία από τις κύριες ανησυχίες σχετικά με το Internet of Things είναι το αρχικό κόστος λόγω του εύρους ζώνης του δικτύου, της αποθήκευσης δεδομένων και της υπολογιστικής ισχύος. Το Edge Computing σε τοπικό επίπεδο μπορεί να εκτελέσει πολλούς υπολογισμούς δεδομένων, οι οποίοι επιτρέπουν στις επιχειρήσεις να αποφασίσουν ποιες υπηρεσίες θα λειτουργούν τοπικά και ποιες θα αποστέλλουν στο cloud, γεγονός το οποίο μειώνει το τελικό κόστος μιας συνολικής λύσης IoT.

Παλιές και σύγχρονες συσκευές: Οι Συσκευές Edge μπορούν να λειτουργήσουν ως μεσάζοντες μεταξύ των παλαιών και σύγχρονων συσκευών. Αυτό επιτρέπει στις παλιές συσκευές να συνδέονται με τις πιο σύγχρονες ή αυτές που χρησιμοποιούν IoT λύσεις και να παρέχουν άμεσα οφέλη από τη λήψη πληροφοριών τόσο από τις παλιές όσο και από τις σύγχρονες.[60]

2.4.4 Χαρακτηριστικά και Θεμελιώδη Μοντέλα Υλοποίησης Τεχνολογιών Νέφους

Η υπηρεσία της υπολογιστικής νέφους χωρίζεται σε κλάσεις με βάση τις δυνατότητες που παρέχει και το μοντέλο υπηρεσίας που χρησιμοποιεί ο πάροχος.

Ενδεικτικά τα κύρια χαρακτηριστικά των τεχνολογιών νέφους είναι τα εξής:

Χρήση κατ' απαίτηση

Βασικό στοιχείο της χρήσης τεχνολογίας νέφους αποτελεί η δημιουργία κατάλληλου περιβάλλοντος και υπηρεσιών, έτσι ώστε ο χρήστης να έχει τη δυνατότητα αξιοποίησης τους, όποτε θελήσει, χωρίς να χρειάζεται ανθρώπινη παρέμβαση αναζήτησης υπηρεσιών.

Ευέλικτη πρόσβαση

Κύριο μέλημα αποτελεί η δυνατότητα πρόσβασης στις υπηρεσίες νέφους μέσα από οποιοδήποτε τερματικό μέσο με παρούσες ικανές δικτυακές υποδομές και επιθυμητά επίπεδα ασφάλειας, ώστε να στηρίζονται επαρκώς οι ζητούμενες υπηρεσίες.

Αξιοποίηση κοινών πόρων

Σε περιβάλλον υπολογιστικής νέφους ο πάροχος αξιοποιεί δεξαμενές από διαφορετικούς πόρους (πχ. υπολογιστική ισχύ, αποθηκευτικό χώρο, λογισμικό) για να εξυπηρετήσει τις ανάγκες των πελατών του. Ανάλογα με τις απαιτήσεις των χρηστών μοιράζονται οι κοινοί πόροι και έτσι η αίσθηση που δίνεται στους χρήστες είναι αυτή των απεριόριστων πόρων, ενώ οποιαδήποτε μεταβολή δεν είναι ορατή στον κοινό χρήστη.

Ελαστικότητα

Με βάση τις ανάγκες, όπως διαμορφώνονται στον χρόνο ή προκύπτουν από συγκεκριμένη χρήση εφαρμογών, υπάρχει η δυνατότητα δυναμικής διάθεσης ή αποδέσμευσης πόρων με διαφανή τρόπο. Με αυτό τον τρόπο οδηγείται ο πάροχος στην καλύτερη αξιοποίηση της υποδομής του, ενώ ο χρήστης διαθέτει τους απαραίτητους πόρους με βάση τις εργασίες προς εκτέλεση.

Μετρούμενη υπηρεσία

Αφορά τη δυνατότητα της καταμέτρησης της χρησιμοποίησης της υπηρεσίας νέφους και της ανάλογης χρέωσης. Η μέτρηση αφορά τόσο στατιστικά στοιχεία χρήσης, όσο και άλλα στοιχεία απαραίτητα για τη διαχείριση της υπηρεσίας και της εκτίμησης της παρεχόμενης ποιότητας.

Αναφορικά με την παρεχόμενη υπηρεσία από τον πάροχο διακρίνουμε:

Λογισμικό σαν Υπηρεσία (Software as a Service - SaaS):

Σε αυτή την περίπτωση οι εφαρμογές προσφέρονται από τον πάροχο και ο χρήστης ούτε διαχειρίζεται ούτε μπορεί να ελέγχει την υποκείμενη υποδομή νέφους και τις δυνατότητες της συγκεκριμένης εφαρμογής. Η συγκεκριμένη υπηρεσία δεν ενδείκνυται για υπηρεσίες πραγματικού χρόνου, που επηρεάζονται από μεταβολές στην κατάσταση της υποδομής και απαιτούν παρεμβάσεις στη διαχείρισή τους. Δημοφιλή παραδείγματα τέτοιων υπηρεσιών αποτελούν το Gmail και η μηχανή αναζήτησης της Google. Ενδεχόμενες πρόσθετες υπηρεσίες είναι: ο Εταιρικές Υπηρεσίες, όπως διαχείριση ροής εργασιών, συνεργατικές υπηρεσίες, εφοδιαστική αλυσίδα, επικοινωνίες, ηλεκτρονική υπογραφή, διαχείριση πελατών, οικονομική

διαχείριση, εργαλεία αναζήτησης και εντοπισμού. ο Εφαρμογές Web 2.0, όπως διαχείριση μεταδεδομένων, κοινωνική δικτύωση, και υπηρεσίες πληροφοριών.

Πλατφόρμα σαν Υπηρεσία (Platform as a Service - PaaS):

Σε αυτή την περίπτωση παρέχεται ολοκληρωμένη πλατφόρμα, όπου ο χρήστης μπορεί να εγκαταστήσει δικές του ή έτοιμες εφαρμογές, χρησιμοποιώντας γλώσσες προγραμματισμού ή εργαλεία που του παρέχει η υπηρεσία. Με αυτό τον τρόπο, ο χρήστης έχει πλήρη έλεγχο της εφαρμογής που εγκαθιστά και σε ορισμένες περιπτώσεις και του περιβάλλοντος που τρέχει η εφαρμογή, αλλά δεν έχει τη δυνατότητα διαχείρισης ή παραμετροποίησης της υποδομής νέφους, όπως το δίκτυο, τους εξυπηρετητές, το λειτουργικό σύστημα ή τον αποθηκευτικό χώρο.

Υποδομή σαν Υπηρεσία (Infrastructure as a Service - IaaS): Ο χρήστης μπορεί να αναπτύξει και να θέσει σε λειτουργία οποιοδήποτε λογισμικό, που μπορεί να συμπεριλαμβάνει ακόμα και λειτουργικά συστήματα μαζί με τυχόν εφαρμογές. Σε αυτή την περίπτωση ο χρήστης ελέγχει τις εφαρμογές που αναπτύσσει και λειτουργεί στην υποδομή, ενώ ενδεχομένως μπορεί να έχει μερικό έλεγχο ακόμα και στον αποθηκευτικό χώρο και τα δικτυακά στοιχεία. Σε αυτή την περίπτωση ενδεικτικές υπηρεσίες αποτελούν υποδομές που φιλοξενούν εξυπηρετητές, παροχή ικανού αποθηκευτικού χώρου και υλικού, πρόσβαση στο διαδίκτυο κ.ά.

Αναφορικά με το περιβάλλον που μπορεί να αναπτυχθεί η τεχνολογία νέφους υπάρχουν τέσσερα κυρίως μοντέλα, συγκεκριμένα:

Δημόσιο νέφος (Public cloud): αναφέρεται σε περιβάλλον που είναι διαθέσιμο προς κάθε χρήστη, ενώ την ευθύνη δημιουργίας, λειτουργίας και συντήρησης τη διατηρεί κάποιος πάροχος με βάση κάποια από τα μοντέλα υπηρεσίας νέφους.

Κοινοτικό νέφος (Community cloud): σε αυτή την περίπτωση οι χρήστες, που έχουν πρόσβαση, πρέπει να είναι μέλη μιας συγκεκριμένης κοινότητας. Παραδείγματα κοινότητας αποτελούν η σχολική κοινότητα, η ακαδημαϊκή κοινότητα, κάποια επαγγελματική ομάδα με κοινά ενδιαφέροντα.

Ιδιωτικό νέφος (Private cloud): Ένα ιδιωτικό νέφος εξυπηρετεί τις ανάγκες μεμονωμένων χρηστών ή οργανισμών και αποτελεί συνήθως ένα αρκετά ελεγχόμενο περιβάλλον.

Υβριδικό νέφος (Hybrid cloud): τα παραπάνω μοντέλα μπορεί να συνδυαστούν, ώστε συνδυασμός παραμέτρων να οδηγεί σε διαφορετικό τύπο. Για παράδειγμα, ένας οργανισμός θα μπορούσε να διατηρεί τις οικονομικές υπηρεσίες του σε ιδιωτικό νέφος και τις υπηρεσίες διάχυσης και ενημέρωσης πελατών σε ένα δημόσιο νέφος. Ένα υβριδικό νέφος αναμένεται να

χρειάζεται προσεκτικότερη διαχείριση και απαιτεί επιπλέον προσπάθεια λόγω του σύνθετου περιβάλλοντος[62][63][64].

2.4.5 Διαχείριση Πόρων και Προγραμματισμός Εργασιών

Η διαχείριση πόρων αποτελεί κρίσιμη λειτουργία και επηρεάζει την επίδοση ενός συστήματος νέφους αναφορικά με τη λειτουργικότητά του, την απόδοσή του και το κόστος. Επιπλέον, ο χρονοπρογραμματισμός χρησιμοποιείται κατά τη διάθεση πόρων ενός συστήματος, όπως κύκλοι κεντρικής μονάδας επεξεργασίας, μνήμη, εναλλακτικός χώρος αποθήκευσης, μονάδες εισόδου/εξόδου και δικτυακοί πόροι. Τέλος, πολιτικές οδηγούν τις όποιες αποφάσεις και τους μηχανισμούς υλοποίησης. Τα δομικά στοιχεία της διαχείρισης πόρων υπολογιστικού νέφους είναι:

Έλεγχος πρόσβασης: αποτρέπει το σύστημα να δεχθεί φόρτο εργασίας, που ενδεχομένως παραβιάζει κανόνες και πολιτικές του. • Ανάθεση χωρητικότητας: χρησιμοποιείται για να διαθέτει τους απαραίτητους πόρους για την ενεργοποίηση μια υπηρεσίας.

Εξισορρόπηση φόρτου: φροντίζει, ώστε ο φόρτος να κατανέμεται ανάλογα με οδηγίες και προδιαγραφές μεταξύ των εξυπηρετητών

Ενεργειακή βελτιστοποίηση: στοχεύει στην ελαχιστοποίηση της δαπανώμενης ενέργειας για κάθε εργασία.

Διασφάλιση ποιότητας: η παροχή επιπέδου ποιότητας υπηρεσίας με βάση συγκεκριμένες χρονικές ή άλλες συνθήκες. Για την υλοποίηση των διαχειριστικών εργαλείων αναπτύσσονται μηχανισμοί, όπως:

Θεωρία ελέγχου: ο μηχανισμός αξιοποιεί ανατροφοδότηση, ώστε να σταθεροποιήσει το σύστημα και να προβλέψει τη συμπεριφορά του σε μεταβατικές συνθήκες.

Μηχανή μάθησης: αξιοποιεί διαδικασία εκμάθησης με βάση κάποιο ενδεχόμενο και το αποτέλεσμά του, χωρίς να απαιτείται γνώση του συστήματος.

Κοινωφελής μηχανισμός: αξιοποιεί μοντέλο επίδοσης με βάση την απόδοση του συστήματος στο επίπεδο του χρήστη και του σχετικού κόστους.

Οικονομικός μηχανισμός: χρησιμοποιεί μηχανισμούς προσφοράς και ζήτησης σε επίπεδο ενιαίας αγοράς με ενδεχόμενες δημοπρασίες.

2.4.6 Ιδιαίτερα Θέματα Υπολογιστικής Νέφους

Με βάση τις ήδη αναπτυσσόμενες προσεγγίσεις και υπηρεσίες προκύπτουν ενδιαφέρουσες τεχνολογικές προκλήσεις που δεν έχουν αντιμετωπιστεί, όπως:

Ασφάλεια, ιδιωτικότητα και εμπιστευτικότητα: η ενότητα μπορεί να επηρεάσει την παροχή υπηρεσίας σε όλα τα επίπεδα, καθότι αξιοποιούνται υποδομές και υπηρεσίες τρίτων που

απαιτούν ικανοποιητική μόχλευση (π.χ. κρυπτογράφηση δεδομένων). Συμπληρωματικά, υφίστανται θέματα νομικής και ρυθμιστικής φύσεως, όπως η τοποθεσία του παρόχου και οι υφιστάμενοι περιορισμοί, ανάλογα με τον τύπο της κάθε προσφερόμενης υπηρεσίας (π.χ. ευαίσθητα δεδομένα να αποθηκεύονται εντός συγκεκριμένων χωρικών ορίων).

Κλείδωμα δεδομένων και προτυποποίηση: η ουσιαστική έλλειψη δια λειτουργικότητας δυσκολεύει τη μετακίνηση από έναν πάροχο υπηρεσιών νέφους σε άλλον, παρότι σχετικός οργανισμός έχει πλέον εστιάσει σε αυτό το στοιχείο (Cloud Computing Interoperability Forum - CCIF)

Διαθεσιμότητα, ανοχή αστοχιών και ανάκαμψη από σφάλματα: Σημαντικό συστατικό στοιχείο στην υιοθέτηση και διατήρηση υπηρεσιών νέφους αποτελούν τα στοιχεία της διαθεσιμότητας της όποιας υπηρεσίας, καθώς και της συνέχειας σε οποιαδήποτε αντίξοη συνθήκη προκύψει. Με τη μορφή συγκεκριμένων συμβολαίων χρήστες και πάροχοι ορίζουν τις παραμέτρους αναφορικά με την παρεχόμενη ποιότητα υπηρεσίας.

Αποδοτική διαχείριση και κατανάλωση ενέργειας: αναφερόμενοι σε μεγάλες μονάδες παροχής υπηρεσιών νέφους απαιτείται η ικανοποιητική διαχείριση του αριθμού των απαιτούμενων λειτουργιών, καθώς και της τεράστιας κατανάλωσης ηλεκτρικής ενέργειας, που επιδρά τόσο στο κόστος όσο και στο περιβάλλον.

2.5 Πλατφόρμα OpenStack

OpenStack[14] είναι μία ελεύθερη (free) και ανοιχτού κώδικά (open source)[15] πλατφόρμα υπολογιστικού νέφους, που ως επι το πλήθος έχει αναπτυχθεί σε υποδομή ως υπηρεσία (infrastructure-as-a-service (IaaS))[16] τόσο σε δημόσια όσο και ιδιωτικά νέφη όπου εικονικοί servers και άλλοι πόροι είναι διαθέσιμοι στους χρήστες. Η πλατφόρμα λογισμικού αποτελείται από αλληλένδετα συστατικά στοιχεία που ελέγχουν ποικίλες πηγές υλικού επεξεργασίας, αποθήκευσης και δικτύωσης σε όλο το κέντρο δεδομένων. Οι χρήστες είτε τα διαχειρίζονται μέσω διαδικτυακών εφαρμογών, επιφανειών εργασίας (dashboards), είτε μέσω web services RestFul που αναφέρονται σ ’ένα αρχιτεκτονικό στυλ λογισμικού που ορίζει ένα σύνολο περιορισμών που πρέπει να χρησιμοποιηθούν για τη δημιουργία υπηρεσιών Web. Το OpenStack

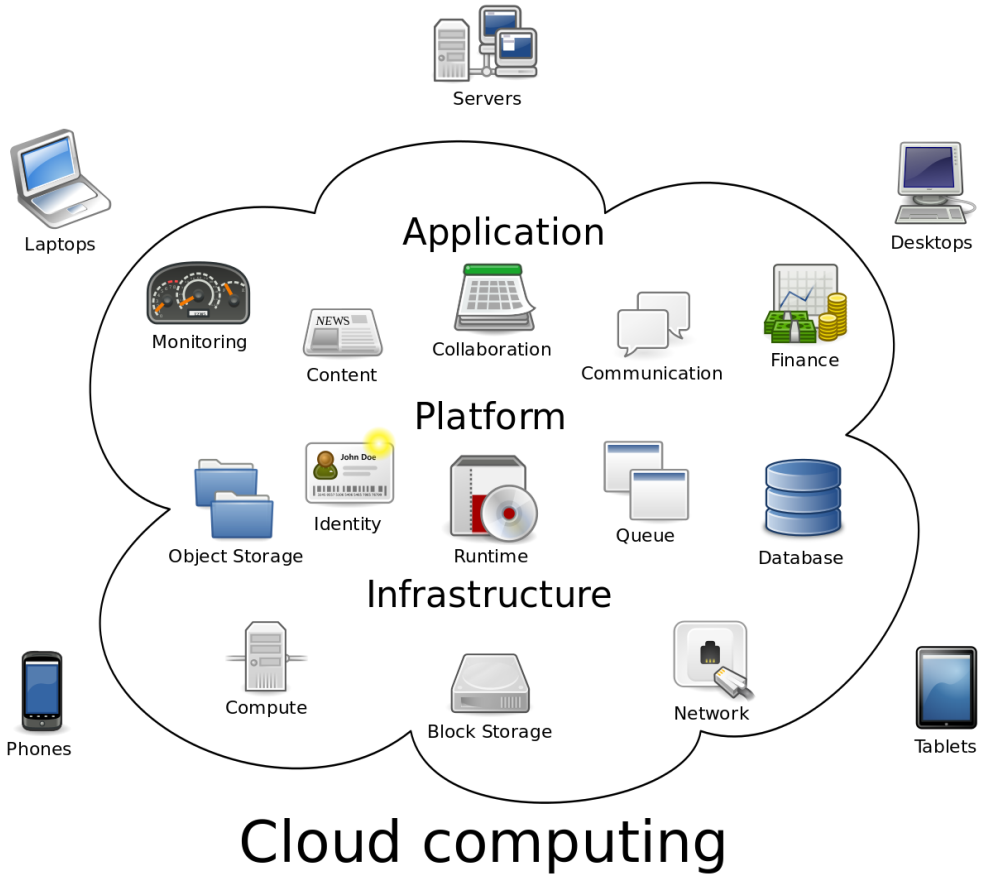
[14] https://en.wikipedia.org/wiki/OpenStack

[15] https://en.wikipedia.org/wiki/Open-source_software

[16] https://en.wikipedia.org/wiki/Cloud_computing#Infrastructure_as_a_service_.28IaaS.29

διαθέτει μια μεγάλη κοινότητα χρηστών, από διάφορους τομείς ενδεικτικά αναφέρονται το BBC,BMW,CERN, NASA,Intel και πολλοί άλλοι κολοσσοί.

Για να δείξουμε σχεδιαστικά πως αξιοποιείται μια πλατφόρμα παραθέτουμε το παρακάτω σχήμα.



Εικόνα 2-5:Cloud Computing[17]

2.5.1 Ιστορικά στοιχεία

Το OpenStack ξεκίνησε το 2010 ως κοινό έργο του Rackspace Hosting μιας εταιρείας cloud computing και της NASA. Από το 2012, διοικείται από το Ίδρυμα OpenStack, μια μη κερδοσκοπική εταιρία που ιδρύθηκε τον Σεπτέμβριο του 2012 για να προωθήσει το λογισμικό OpenStack και την κοινότητά του. Πλέον περισσότερες από 500 εταιρείες έχουν μπει στο συγκεκριμένο έργο. Στόχος ήταν "να παράγουν την μεγαλύτερη παρούσα πλατφόρμα υπολογιστών ανοιχτού κώδικα Cloud Computing που θα καλύπτει τις ανάγκες των δημόσιων και

[17] https://en.wikipedia.org/wiki/Cloud_computing#/media/File:Cloud_computing.svg

ιδιωτικών σύννεφων ανεξάρτητα από το μέγεθος, με την απλή υλοποίησή τους και τη μαζική κλιμάκωση"

Το πρόγραμμα OpenStack προορίζεται να βοηθήσει τους οργανισμούς να προσφέρουν υπηρεσίες cloud-computing που εκτελούνται σε συγκεκριμένους hardware πόρους. Η πρώτη επίσημη κυκλοφορία της κοινότητας, με την κωδική ονομασία Austin, εμφανίστηκε τρεις μήνες αργότερα στις 21 Οκτωβρίου 2010, με σχέδια να προσφέρουν τακτικές ενημερώσεις του λογισμικού κάθε λίγους μήνες. Ο πρώτος κώδικας προήλθε από την πλατφόρμα Nebula της NASA καθώς και από την πλατφόρμα Cloud Files του Rackspace. Η αρχική αρχιτεκτονική νέφους σχεδιάστηκε από το NASA Ames Web Manager, Megan A. Eskey και ήταν μια αρχιτεκτονική ανοιχτού κώδικα 2009 που ονομάζεται OpenNASA v2.0. Οι στοίβες σύννεφων και οι ανοικτές ενότητες στοίβας συγχωνεύθηκαν και απελευθερώθηκαν ως ανοιχτή πηγή από την ομάδα Nedula της NASA σε συνεννόηση με το Rackspace.

Το 2011, οι προγραμματιστές της διανομής Ubuntu Linux υιοθέτησαν το OpenStack με μια μη υποστηριζόμενη προεπισκόπηση της εξέλιξης του OpenStack "Bexar" για το Ubuntu 11.04 "Natty Narwhal".Στη συνέχεια, ο χορηγός της Canonical της Ubuntu εισήγαγε πλήρη υποστήριξη για σύννεφα OpenStack, ξεκινώντας με την έκδοση Cactus του OpenStack.

Το 2012, η Red Hat ανακοίνωσε μια δική της προεπισκόπηση της διανομής OpenStack.Τον Δεκέμβριο του 2013, η Oracle ανακοίνωσε ότι προσχώρησε στο OpenStack ως Χορηγός και σχεδίαζε να φέρει το OpenStack στην Oracle Solaris, στο Oracle Linux και σε πολλά από τα προϊόντα της. Από τον Μάρτιο του 2015, η NASA εξακολουθεί να χρησιμοποιεί το ιδιωτικό cloud OpenStack και έχει RFPs (είναι ένα έγγραφο που απαιτεί την υποβολή προτάσεων, που συχνά πραγματοποιούνται μέσω διαδικασίας υποβολής προσφορών, από έναν οργανισμό ή εταιρεία που ενδιαφέρεται να προμηθεύσει ένα εμπόρευμα, μια υπηρεσία ή ένα πολύτιμο περιουσιακό στοιχείο, σε δυνητικούς προμηθευτές να υποβάλουν επιχειρηματικές προτάσεις) για δημόσια cloud υποστήριξη OpenStack .

2.5.2 Υποσυστήματα του OpenStack

Παρακάτω θα παρατίθενται συνοπτικά όλες τις υπηρεσίες υποσυστήματα και project που αναπτύσσονται στην πλατφόρμα OpenStack[53][54].

Υπολογισμός Compute (Nova)

Το Nova είναι ένα έργο του OpenStack που παρέχει έναν τρόπο παροχής υπολογιστικών instances στιγμιότυπα (εικονικοί servers). Η Nova υποστηρίζει τη δημιουργία εικονικών μηχανών, servers baremetal και έχει περιορισμένη υποστήριξη . Λειτουργεί ως ένα σύνολο από

δαίμονες πάνω από υπάρχοντες διακομιστές Linux για να παρέχει αυτή την υπηρεσία και είναι γραμμένη σε Python. Χρησιμοποιεί πολλές εξωτερικές βιβλιοθήκες της Python, όπως το Eventlet (βιβλιοθήκη ταυτόχρονης δικτύωσης), το Kombu (πλαίσιο ανταλλαγής μηνυμάτων AMQP) και το SQLAlchemy (SQL toolkit και Object Relational Mapper).

Το Nova έχει σχεδιαστεί ώστε να είναι οριζόντια κλιμακωτή (horizontally scalable). Αντί να αλλάζετε σε μεγαλύτερους διακομιστές, προμηθεύεστε περισσότερους διακομιστές και απλά εγκαταστήσετε υπηρεσίες με τις ίδιες ρυθμίσεις. Λόγω της εκτεταμένης ενσωμάτωσής της σε επιχειρηματικές υποδομές, την παρακολούθηση της απόδοσης του OpenStack εν γένει και ειδικότερα της απόδοσης της Nova, η κλιμάκωση έχει καταστεί όλο και πιο σημαντικό ζήτημα. Η παρακολούθηση της απόδοσης σε επίπεδο end-to-end απαιτεί μετρήσεις παρακολούθησης από τα Nova, Keystone, Neutron, Cinder, Swift και άλλες υπηρεσίες, εκτός από την παρακολούθηση του RabbitMQ που χρησιμοποιείται από τις υπηρεσίες OpenStack για τη μετάδοση μηνυμάτων. Όλες αυτές οι υπηρεσίες δημιουργούν τα δικά τους αρχεία καταγραφής, τα οποία, ειδικά σε υποδομές σε επίπεδο επιχειρήσεων, πρέπει επίσης να παρακολουθούνται.

Δικτύωση Networking (Neutron)

Το Neutron είναι ένα έργο project OpenStack που παρέχει "συνδεσιμότητα δικτύου ως υπηρεσία" μεταξύ συσκευών διασύνδεσης (π.χ. vNIC) που διαχειρίζονται άλλες υπηρεσίες OpenStack (π.χ. nova). Εφαρμόζει το API OpenStack Networking.

Διαχειρίζεται όλες τις πτυχές δικτύωσης για τις υποδομές εικονικής δικτύωσης (VNI) και τις πτυχές του στρώματος πρόσβασης της υποδομής φυσικής δικτύωσης (Physical Networking Infrastructure - PNI) στο περιβάλλον OpenStack. Το OpenStack Networking επιτρέπει στα έργα να δημιουργούν προηγμένες τοπολογίες εικονικών δικτύων, οι οποίες μπορεί να περιλαμβάνουν υπηρεσίες όπως ένα τείχος προστασίας και ένα εικονικό ιδιωτικό δίκτυο (VPN).

Το Neutron επιτρέπει ειδικές στατικές διευθύνσεις IP ή DHCP. Επιτρέπει επίσης τις κυψέλες διευθύνσεων IP να επιτρέπουν τη δυναμική επαναδιάταξη της κυκλοφορίας.

Οι χρήστες μπορούν να χρησιμοποιήσουν τεχνολογίες δικτύου (SDN) που έχουν καθοριστεί από το λογισμικό, όπως το OpenFlow, για να υποστηρίξουν την πολυεθνική μίσθωση και την κλίμακα. Η δικτύωση OpenStack μπορεί να αναπτύξει και να διαχειριστεί πρόσθετες υπηρεσίες δικτύου - όπως συστήματα ανίχνευσης εισβολών (IDS), εξισορρόπηση φορτίου, τείχη προστασίας και εικονικά ιδιωτικά δίκτυα (VPN).

Αποκλεισμός χώρου αποθήκευσης Block storage (Cinder)

Το Cinder είναι η υπηρεσία αποθήκευσης του OpenStack Block για την παροχή τόμων σε εικονικές μηχανές Nova, κεντρικούς υπολογιστές, καταχωρητές και πολλά άλλα. Μερικοί από τους στόχους του Cinder πρέπει να είναι:

Αρχιτεκτονική βασισμένη σε υποσυστήματα: Να προστίθενται άμεσα και γρήγορα νέες συμπεριφορές-νέες τάσεις.

Υψηλής Διαθεσιμότητας: να γίνεται κλιμάκωση σε πολύ υψηλούς φόρτους εργασίας workloads.

Ανθεκτική σε σφάλματα: Μεμονωμένες διαδικασίες αποφεύγουν τις αποτυχημένες αστοχίες

Να είναι ανακτήσιμοι: Οι αποτυχίες πρέπει να είναι εύκολο να διαγνωστούν και να διορθωθούν.

Ανοιχτά πρότυπα: Να είναι μια εφαρμογή αναφοράς από κάποια κοινότητα που καθοδηγεί ένα api.

Οι τόμοι Cinder παρέχουν μόνιμη αποθήκευση σε εικονικές μηχανές guest - γνωστές ως στιγμιότυπα instances, οι οποίες διαχειρίζονται το λογισμικό OpenStack Compute. Το Cinder μπορεί επίσης να χρησιμοποιηθεί ανεξάρτητα από άλλες υπηρεσίες OpenStack ως ανεξάρτητη αποθήκευση που έχει καθοριστεί από το λογισμικό. Το σύστημα αποθήκευσης μπλοκ διαχειρίζεται τη δημιουργία, αναπαραγωγή, διαχείριση στιγμιότυπων, σύνδεση και αποσύνδεση των συσκευών σε διακομιστές.

Ταυτότητα Identity (Keystone)

Το Keystone είναι μια υπηρεσία OpenStack που παρέχει ένα API έλεγχο ταυτότητας πελάτη, ανακάλυψη υπηρεσιών και εξουσιοδότηση πολλαπλών μισθωτών μέσω της εφαρμογής του API ταυτότητας του OpenStack. Είναι το κοινό σύστημα ελέγχου ταυτότητας σε όλο το λειτουργικό σύστημα cloud. Το Keystone μπορεί να ενσωματωθεί με υπηρεσίες καταλόγου όπως το LDAP. Υποστηρίζει τυποποιημένες πιστοποιήσεις ονόματος χρήστη και κωδικού πρόσβασης, συστήματα που βασίζονται σε διακριτικά σήματα και αρχεία τύπου AWS (δηλαδή υπηρεσίες Amazon Web Services). Ο κατάλογος υπηρεσιών OpenStack Keystone Service επιτρέπει στους πελάτες API να ανακαλύπτουν δυναμικά και να περιηγηθούν στις υπηρεσίες cloud.

Εικόνα Image (Glance)

Το έργο της υπηρεσίας εικόνας (προβολής) παρέχει μια υπηρεσία όπου οι χρήστες μπορούν να μεταφορτώσουν και να ανακαλύψουν στοιχεία δεδομένων που προορίζονται να χρησιμοποιηθούν με άλλες υπηρεσίες. Συνήθως περιλαμβάνει ορισμούς εικόνων και μεταδεδομένων.

Εικόνες Images

Οι υπηρεσίες εικόνας περιλαμβάνουν την ανακάλυψη, την εγγραφή και την ανάκτηση εικόνων εικονικής μηχανής (VM). Το Glance έχει ένα RESTful API που επιτρέπει την αναζήτηση των μεταδεδομένων εικόνας VM καθώς και την ανάκτηση της πραγματικής εικόνας. Οι εικόνες VM που διατίθενται μέσω της οθόνης μπορούν να αποθηκευτούν σε διάφορες τοποθεσίες, από απλά συστήματα αρχείων έως συστήματα αντικειμένων αποθήκευσης όπως το project OpenStack Swift.

Ορισμοί μεταδεδομένων

Το Glance φιλοξενεί έναν κατάλογο μεταδεδομένων. Αυτό παρέχει στην κοινότητα του OpenStack έναν τρόπο να προσδιορίζει με προγραμματισμό διάφορα ονόματα κλειδιών μεταδεδομένων και έγκυρες τιμές που μπορούν να εφαρμοστούν σε πόρους του OpenStack

Αποθήκευση αντικειμένων (Swift)

Το Swift είναι ένα κατανεμημένο, τελικά σταθερό αποθηκευτικό μέσο αντικειμένων / μπλοκ. Το έργο OpenStack Object Store, γνωστό ως Swift, προσφέρει λογισμικό αποθήκευσης cloud έτσι ώστε να μπορείτε να αποθηκεύσετε και να ανακτήσετε πολλά δεδομένα με ένα απλό API. Είναι κατασκευασμένο για κλίμακα και βελτιστοποιείται για αντοχή, διαθεσιμότητα και ταυτότητα σε ολόκληρο το σύνολο δεδομένων. Το Swift είναι ιδανικό για την αποθήκευση μη δομημένων δεδομένων που μπορούν να αναπτυχθούν χωρίς σύνδεση.

Πίνακας ελέγχου (Horizon)

Το Horizon είναι η κανονική εφαρμογή του Dashboard του OpenStack, το οποίο παρέχει ένα web-based user interface για υπηρεσίες OpenStack, όπως Nova, Swift, Keystone, κλπ. ο Horizon διαθέτει τρεις κεντρικούς πίνακες ελέγχου, ένα "Πίνακα ελέγχου χρήστη", ένα "Πίνακα ελέγχου συστήματος" και ένα ταμπλό "Ρυθμίσεις". Μεταξύ αυτών των τριών καλύπτουν τις βασικές εφαρμογές του OpenStack και προσφέρουν την Υποστήριξη Core. Η εφαρμογή Horizon συνοδεύεται επίσης από μια σειρά από αφαιρέσεις API για τα βασικά έργα του OpenStack προκειμένου να παράσχει ένα σταθερό και σταθερό σύνολο επαναχρησιμοποιούμενων μεθόδων για τους προγραμματιστές. Χρησιμοποιώντας αυτές τις περιλήψεις, οι προγραμματιστές που εργάζονται στο Horizon δεν χρειάζεται να εξοικειωθούν στενά με τα API κάθε έργου OpenStack

Ενορχήστρωση Orchestration (Heat)

Το heat είναι μια υπηρεσία για την ενορχήστρωση πολλαπλών σύνθετων εφαρμογών νέφους με χρήση προτύπων, μέσω ενός API REST του OpenStack και ενός API Query συμβατό με το CloudFormation.

Ροή εργασίας Workflow (Mistral)

Το Mistral είναι μια υπηρεσία που διαχειρίζεται ροές εργασίας. Ο χρήστης συνήθως γράφει μια ροή εργασίας χρησιμοποιώντας τη γλώσσα ροής εργασίας που βασίζεται στο YAML και μεταφορτώνει τον ορισμό της ροής εργασίας στο Mistral μέσω του REST API. Στη συνέχεια, ο χρήστης μπορεί να ξεκινήσει αυτή τη ροή εργασίας με το χέρι μέσω του ίδιου API ή να διαμορφώσει έναν trigger για να ξεκινήσει τη ροή εργασίας σε κάποιο συμβάν.

Τηλεμετρία Telemetry (Ceilometer)

Το OpenStack (Ceilometer) είναι μία υπηρεσία που παρέχει ένα ενιαίο σημείο επαφής για τα συστήματα χρέωσης, παρέχοντας όλους τους μετρητές που χρειάζονται για να καθορίσουν την τιμολόγηση πελατών, σε όλα τα τρέχοντα και μελλοντικά συστατικά του OpenStack. Η παράδοση μετρητών είναι ανιχνεύσιμη και ελέγξιμη, οι μετρητές πρέπει να είναι εύκολα επεκτάσιμοι για την υποστήριξη νέων έργων και οι πράκτορες που συλλέγουν δεδομένα πρέπει να είναι ανεξάρτητοι από το συνολικό σύστημα.

Βάση δεδομένων Database (Trove)

Το Trove είναι μια σχεσιακή παροχή βάσης δεδομένων ως παροχή υπηρεσιών (as a service) και μια μη σχεσιακή μηχανή βάσης δεδομένων.

Elastic map reduce (Sahara)

Το Sahara αποτελεί ένα συστατικό στοιχείο για την εύκολη και γρήγορη δημιουργία clustersHadoop (μια συλλογή βοηθητικών προγραμμάτων λογισμικού ανοιχτού κώδικα που διευκολύνουν τη χρήση ενός δικτύου πολλών υπολογιστών για την επίλυση προβλημάτων που αφορούν τεράστια ποσά δεδομένων και υπολογισμών). Οι χρήστες θα καθορίσουν διάφορες παραμέτρους όπως τον αριθμό έκδοσης του Hadoop, τον τύπο τοπολογίας συμπλέγματος, λεπτομέρειες του κόμβου (καθορίζοντας χώρο στο δίσκο, ρυθμίσεις CPU και RAM) και άλλα. Αφού ο χρήστης παράσχει όλες τις παραμέτρους, το Sahara αναπτύσσει το σύμπλεγμα σε λίγα λεπτά. Το Shahara παρέχει επίσης μέσα για την κλιμάκωση ενός προϋπάρχοντος cluster Hadoop προσθέτοντας και αφαιρώντας τους συνεργαζόμενους κόμβους (on demand) κατ' απαίτηση.

Bare metal (Ironic)

Το Ironic είναι ένα έργο OpenStack που προβλέπει μη μεταλλικά μηχανήματα αντί εικονικών μηχανών. Ξεκίνησε αρχικά από τον οδηγό Nova Baremetal και έχει εξελιχθεί σε ξεχωριστό έργο. Είναι καλύτερα να θεωρείται ως API μεταλλικού hypervisor και ένα σύνολο από plugins που αλληλοεπιδρούν με τους μη μεταλλικούς hypervisors.

Messaging Μηνύματα (Zaqar)

Το Zaqar είναι μια υπηρεσία πολλαπλών κατόχων μηνυμάτων cloud για προγραμματιστές Ιστού. Η υπηρεσία διαθέτει ένα πλήρες RESTful API, το οποίο οι προγραμματιστές μπορούν να χρησιμοποιήσουν για να στέλνουν μηνύματα μεταξύ διαφόρων εξαρτημάτων των SaaS και των κινητών τους εφαρμογών χρησιμοποιώντας μια ποικιλία μοντέλων επικοινωνίας. Αυτό το API είναι μια αποτελεσματική μηχανή ανταλλαγής μηνυμάτων σχεδιασμένη με γνώμονα την επεκτασιμότητα και την ασφάλεια.

Σύστημα κοινής χρήσης αρχείων Shared file system (Manila)

Το σύστημα κοινόχρηστου αρχείου OpenStack (Manila) παρέχει ένα ανοικτό API για τη διαχείριση των κοινόχρηστων στοιχείων κάτω από την επίβλεψη ενός πλαίσιο (framework). Τα τυπικά πρωτόκολλα περιλαμβάνουν τη δυνατότητα δημιουργίας, διαγραφής και παροχής / άρνησης πρόσβασης σε μια κοινή χρήση και μπορούν να χρησιμοποιηθούν ανεξάρτητα ή σε διάφορα διαφορετικά περιβάλλοντα δικτύου. Οι συσκευές αποθήκευσης υποστηρίζονται από την EMC, NetApp, HP, IBM, Oracle, Quabyte, INFINIDAT και Hitachi Data Systems , καθώς και τεχνολογίες συστημάτων αρχείων όπως το Red Hat GlusterFS.

DNS (Designate)

Το Designate είναι ένα πολυλειτουργικό REST API για τη διαχείριση DNS. Αυτό το στοιχείο παρέχει το DNS ως Υπηρεσία και είναι συμβατό με πολλές τεχνολογίες backend, συμπεριλαμβανομένων των PowerDNS και BIND. Δεν παρέχει μια υπηρεσία DNS ως έχει, ο σκοπός της οποίας είναι η διασύνδεση με υπάρχοντες διακομιστές DNS για τη διαχείριση των ζωνών DNS σε βάση ανά ενοικιαστή.

Διαχειριστής κλειδιού Key manager (Barbican)

Το Barbican είναι ένα API REST σχεδιασμένο για την ασφαλή αποθήκευση, παροχή και διαχείριση μυστικών. Σκοπός του είναι να είναι χρήσιμο για όλα τα περιβάλλοντα, συμπεριλαμβανομένων των μεγάλων εφήμερων σύννεφων (ephemeral Clouds).

Αναζήτηση (Searchlight)

Το Searchlight παρέχει προηγμένες και συνεπείς δυνατότητες αναζήτησης σε διάφορες υπηρεσίες cloud του OpenStack.

Ενορχήστρωση Container (Magnum)

Το Magnum είναι μια υπηρεσία API του OpenStack που αναπτύχθηκε από την ομάδα OpenStack Containers, καθιστώντας τους κινητήρες ενορχηστρώσεων Container όπως το Docker

Swarm, το Kubernetes και το Apache Mesos διαθέσιμοι ως πόροι πρώτης κατηγορίας στο OpenStack. Το Magnum χρησιμοποιεί τη θερμότητα για να ενορχηστρώσει μια εικόνα OS που περιέχει Docker και Kubernetes και τρέχει αυτή την εικόνα είτε σε εικονικές μηχανές είτε σε γυμνό μέταλλο σε διαμόρφωση συμπλεγμάτων.

Root Cause Analysis (Vitrage)

Το Vitrage είναι η υπηρεσία OpenStack RCA (Root Cause Analysis) για την οργάνωση, την ανάλυση και την επέκταση των συναγερμών και συμβάντων OpenStack, δίνοντας πληροφορίες σχετικά με τη βασική αιτία των προβλημάτων και αφαιρώντας την ύπαρξή τους πριν εντοπιστούν άμεσα

Ενέργειες συναγερμού που βασίζονται σε κανόνες (Aodh)

Αυτή η υπηρεσία συναγερμών καθιστά δυνατή την ενεργοποίηση ενεργειών βάσει καθορισμένων κανόνων κατά δεδομένων μετρήσεων ή συμβάντων που συλλέγονται από το Ceilometer ή το Gnocchi[55]

2.5.3 Συμβατότητα με άλλα API νέφους

Το OpenStack δεν επιδιώκει τη συμβατότητα με API άλλων σύννεφων. Ωστόσο, υπάρχει κάποιος βαθμό συμβατότητας που καθοδηγείται από διάφορα μέλη της κοινότητας OpenStack για τα οποία είναι σημαντικά αυτό είναι σημαντικό.

- Το έργο EC2 API στοχεύει να παρέχει συμβατότητα με το Amazon EC2 .
- Το έργο GCE API στοχεύει να παρέχει συμβατότητα με το Google Compute Engine .

2.5.4 Εφαρμογές (Appliances)

Το OpenStack Appliance είναι το όνομα που δόθηκε στο λογισμικό που μπορεί να υποστηρίξει την πλατφόρμα υπολογιστικού νέφους OpenStack είτε σε φυσικές συσκευές όπως διακομιστές ή εικονικές μηχανές ή σε συνδυασμό των δύο. Συνήθως ένα appliance λογισμικού είναι ένα σύνολο δυνατοτήτων λογισμικού που μπορεί να λειτουργήσει χωρίς ένα λειτουργικό σύστημα. Επομένως, πρέπει να περιέχουν αρκετές από τις βασικές συνιστώσες του λειτουργικού συστήματος για να λειτουργούν. Ως εκ τούτου, ένας αυστηρός ορισμός μπορεί να είναι: μια εφαρμογή που έχει σχεδιαστεί για να προσφέρει δυνατότητα OpenStack χωρίς την ανάγκη ενός υποκείμενου λειτουργικού συστήματος.

2.5.5 Προκλήσεις κατά την εφαρμογή του

Το OpenStack είναι μια πολύπλοκη οντότητα και όσοι το υιοθετούν αντιμετωπίζουν μια σειρά προκλήσεων κατά την προσπάθεια υλοποίησης του OpenStack σε έναν οργανισμό. Για πολλούς οργανισμούς που προσπαθούν να υλοποιήσουν τα δικά τους έργα, ένα βασικό ζήτημα είναι η έλλειψη διαθέσιμων δεξιοτήτων

Οι πέντε προκλήσεις που θα αντιμετωπίσει κάθε οργανισμός που επιθυμεί να αναπτύξει το OpenStack

Προκλήσεις εγκατάστασης

Το OpenStack είναι μια σουίτα εργαλείων και όχι ένα μόνο προϊόν και επειδή κάθε μία από τις διάφορες εφαρμογές πρέπει να ρυθμιστεί ώστε να ταιριάζει στις απαιτήσεις του χρήστη, η εγκατάσταση είναι πολύπλοκη και απαιτεί μια σειρά συμπληρωματικών ρυθμίσεων για βέλτιστη απόδοση .

Τεκμηρίωση

Η τεκμηρίωση 25, παραπάνω από 25 έργων είναι πραγματική πρόκληση και πάντα είναι σε συνάρτηση με την φύση μίας τεκμηρίωσης προϊόντος ανοιχτού κώδικα.

Αναβάθμιση του OpenStack

Ένας από τους κύριους στόχους της χρήσης της υποδομής τύπου cloud είναι ότι προσφέρει στους χρήστες της όχι μόνο υψηλή αξιοπιστία αλλά και υψηλή διαθεσιμότητα κάτι που οι δημόσιοι προμηθευτές νέφους θα προσφέρουν στις συμφωνίες επιπέδου υπηρεσιών. Λόγω της προσέγγισης ανάπτυξης πολλαπλών έργων του OpenStack, η πολυπλοκότητα που συνδέεται με το συγχρονισμό των διαφόρων έργων κατά τη διάρκεια μιας αναβάθμισης μπορεί να σημαίνει ότι ο χρόνος διακοπής είναι αναπόφευκτος.

Μακροπρόθεσμη υποστήριξη

Είναι αρκετά συνηθισμένο για μια επιχείρηση να συνεχίσει να χρησιμοποιεί προηγούμενη έκδοση λογισμικού για κάποιο χρονικό διάστημα μετά την αναβάθμισή της. Οι λόγοι για αυτό είναι αρκετά προφανείς και αναφέρονται παραπάνω. Ωστόσο, υπάρχει ελάχιστο κίνητρο για τους προγραμματιστές σε ένα έργο ανοιχτού κώδικα για την παροχή υποστήριξης για τον αντικατασταθέντα κώδικα. Επιπλέον, το ίδιο το OpenStack έχει τυπικά διακόψει την υποστήριξη για ορισμένες παλιές κυκλοφορίες. Με βάση τις παραπάνω προκλήσεις, η πιο κατάλληλη διαδρομή για έναν οργανισμό που επιθυμεί να εφαρμόσει το OpenStack θα ήταν να

πάει με έναν προμηθευτή και να προμηθευθεί ένα OpenStack appliance ή σε κάποιον προμηθευτή για να του δώσει μία διανομή αυτού.[]

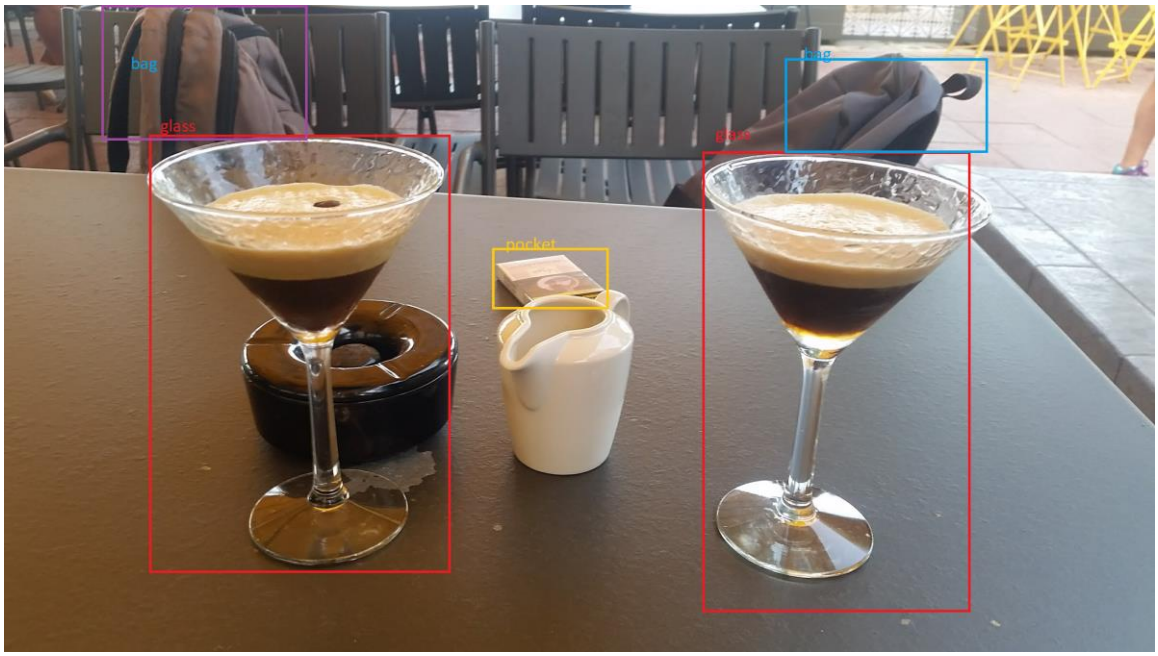
2.6 Αναγνώριση Αντικειμένων

2.6.1 Ανίχνευση Αντικειμένου

Η ανίχνευση αντικειμένων είναι μια τεχνολογία της πληροφορικής που σχετίζεται με την όραση και την επεξεργασία των εικόνων με την ανίχνευση περιπτώσεων αντικειμένων μιας συγκεκριμένης κατηγορίας (όπως άνθρωποι, κτίρια ή αυτοκίνητα) σε ψηφιακές εικόνες και βίντεο. Πολύ καλά εξειδικευμένοι τομείς στην ανίχνευση αντικειμένων περιλαμβάνουν ανίχνευση προσώπου και ανίχνευση πεζών. Η ανίχνευση αντικειμένων έχει εφαρμογές σε πολλούς τομείς , συμπεριλαμβανομένης της ανάκτησης εικόνων και της βιντεοεπιτήρησης.

Οι εφαρμογές που έχει η αναγνώριση αντικειμένων μπορεί να είναι περιπτώσεις ηλεκτρονικής όρασης, όπως σχολιασμός εικόνας, αναγνώριση δραστηριότητας, ανίχνευση προσώπου. Χρησιμοποιείται επίσης για την παρακολούθηση αντικειμένων, όπως για παράδειγμα η παρακολούθηση μιας μπάλας κατά τη διάρκεια ενός ποδοσφαιρικού αγώνα ή η παρακολούθηση ενός ατόμου σε ένα βίντεο.

Η ιδέα υλοποιείται ως εξής, κάθε κατηγορία αντικειμένων έχει τα δικά της ειδικά χαρακτηριστικά που βοηθούν στην ταξινόμηση της κατηγορίας- για παράδειγμα όλοι οι κύκλοι είναι στρογγυλοί. Η ανίχνευση κατηγορίας αντικειμένων χρησιμοποιεί αυτά τα ειδικά χαρακτηριστικά. Για παράδειγμα, όταν αναζητάτε κύκλους, αναζητούνται αντικείμενα που βρίσκονται σε συγκεκριμένη απόσταση από ένα σημείο (δηλ. Το κέντρο). Ομοίως, όταν ψάχνετε για τετράγωνα, χρειάζονται αντικείμενα που είναι κάθετα στις γωνίες και έχουν ίσα πλάτη πλευρά. Μια παρόμοια προσέγγιση χρησιμοποιείται για την αναγνώριση προσώπου όπου μπορούν να βρεθούν τα μάτια, η μύτη και τα χείλη και χαρακτηριστικά όπως το χρώμα του δέρματος και η απόσταση μεταξύ των ματιών.



Εικόνα 2-6:Object Detection

Η αναγνώριση αντικειμένων είναι ένας γενικός όρος που περιγράφει μια συλλογή σχετικών εργασιών όρασης υπολογιστή που περιλαμβάνουν την αναγνώριση αντικειμένων σε ψηφιακές φωτογραφίες. Η ταξινόμηση εικόνας περιλαμβάνει την πρόβλεψη της κλάσης ενός αντικειμένου σε μια εικόνα. Ο εντοπισμός αντικειμένων αναφέρεται στον εντοπισμό της θέσης ενός ή περισσότερων αντικειμένων σε μια εικόνα και σχεδίασης αφθονίας πλαισίου γύρω από την έκτασή τους. Η ανίχνευση αντικειμένων συνδυάζει αυτές τις δύο εργασίες και εντοπίζει και ταξινομεί ένα ή περισσότερα αντικείμενα σε μια εικόνα. Όταν ένας χρήστης ή επαγγελματίας αναφέρεται στην «αναγνώριση αντικειμένων», συχνά εννοούν την «ανίχνευση αντικειμένων».

Θα χρησιμοποιούμε τον όρο αναγνώριση αντικειμένων ευρέως για να συμπεριλάβουμε τόσο την ταξινόμηση εικόνας (μια εργασία που απαιτεί έναν αλγόριθμο για τον προσδιορισμό των κατηγοριών αντικειμένων που υπάρχουν στην εικόνα) όσο και την ανίχνευση αντικειμένων (μια εργασία που απαιτεί έναν αλγόριθμο για τον εντοπισμό όλων των αντικειμένων που υπάρχουν στο εικόνα.[4][5]

Ως εκ τούτου, μπορούμε να διακρίνουμε μεταξύ αυτών των τριών εργασιών όρασης υπολογιστή:

Ταξινόμηση εικόνας - (Image Classification): Προβλέψτε τον τύπο ή την κλάση ενός αντικειμένου σε μια εικόνα.

- Είσοδος: Μια εικόνα με ένα αντικείμενο, όπως μια φωτογραφία.
- Έξοδος: Μια ετικέτα κλάσης (π.χ. ένας ή περισσότεροι ακέραιοι που αντιστοιχίζονται σε ετικέτες κλάσης).

Εντοπισμός αντικειμένων – (Object Localization): Εντοπίστε την παρουσία αντικειμένων σε μια εικόνα και υποδείξτε τη θέση τους με ένα πλαίσιο οριοθέτησης.

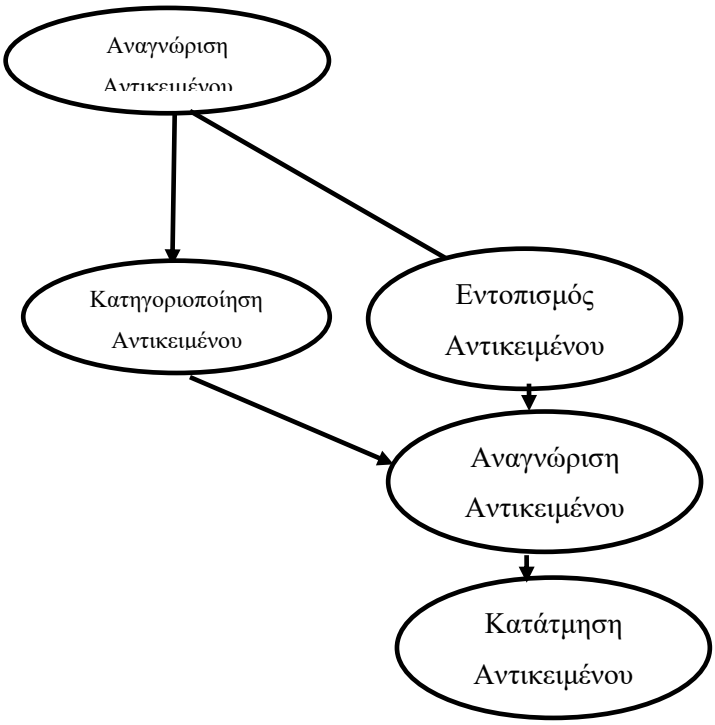
- Είσοδος: Μια εικόνα με ένα ή περισσότερα αντικείμενα, όπως μια φωτογραφία.
- Έξοδος: Ένα ή περισσότερα κουτιά οριοθέτησης (π.χ. καθορίζονται από ένα σημείο, πλάτος και ύψος).

Ανίχνευση αντικειμένων – (Object Detection): Εντοπίστε την παρουσία αντικειμένων με ένα πλαίσιο οριοθέτησης και τύπους ή κλάσεις των αντικειμένων που βρίσκονται σε μια εικόνα.

- Είσοδος: Μια εικόνα με ένα ή περισσότερα αντικείμενα, όπως μια φωτογραφία.
- Έξοδος: Ένα ή περισσότερα κουτιά οριοθέτησης (π.χ. καθορίζονται από ένα σημείο, πλάτος και ύψος) και μια ετικέτα κλάσης για κάθε πλαίσιο οριοθέτησης.

Μία περαιτέρω επέκταση σε αυτήν την ανάλυση των εργασιών όρασης υπολογιστή είναι η κατηγοριοποίηση αντικειμένων, που ονομάζεται επίσης «κατηγοριοποίηση παρουσίας αντικειμένου» ή «σημασιολογική κατηγοριοποίηση», όπου οι παρουσίες αναγνωρισμένων αντικειμένων υποδεικνύονται επισημαίνοντας τα συγκεκριμένα εικονοστοιχεία του αντικειμένου αντί για ένα γενικό πλαίσιο οριοθέτησης.

Από αυτήν την ανάλυση, μπορούμε να δούμε ότι η αναγνώριση αντικειμένων αναφέρεται σε μια σειρά εργασιών όρασης υπολογιστή.



Εικόνα 2-7:Επισκόπηση αντικειμένων αναγνώρισης αντικειμένων Computer Vision

Οι περισσότερες από τις πρόσφατες καινοτομίες στα προβλήματα αναγνώρισης εικόνας έχουν προκύψει ως μέρος της συμμετοχής στις εργασίες του ILSVRC. Πρόκειται για έναν ετήσιο ακαδημαϊκό διαγωνισμό με ξεχωριστή πρόκληση για καθέναν από αυτούς τους τρεις τύπους προβλημάτων, με σκοπό την προώθηση ανεξάρτητων και ξεχωριστών βελτιώσεων σε κάθε επίπεδο που μπορούν να αξιοποιηθούν ευρύτερα. Για παράδειγμα, δείτε τη λίστα των τριών αντίστοιχων τύπων εργασιών που ελήφθησαν από τη δημοσίευση αξιολόγησης ILSVRC 2015 [48].

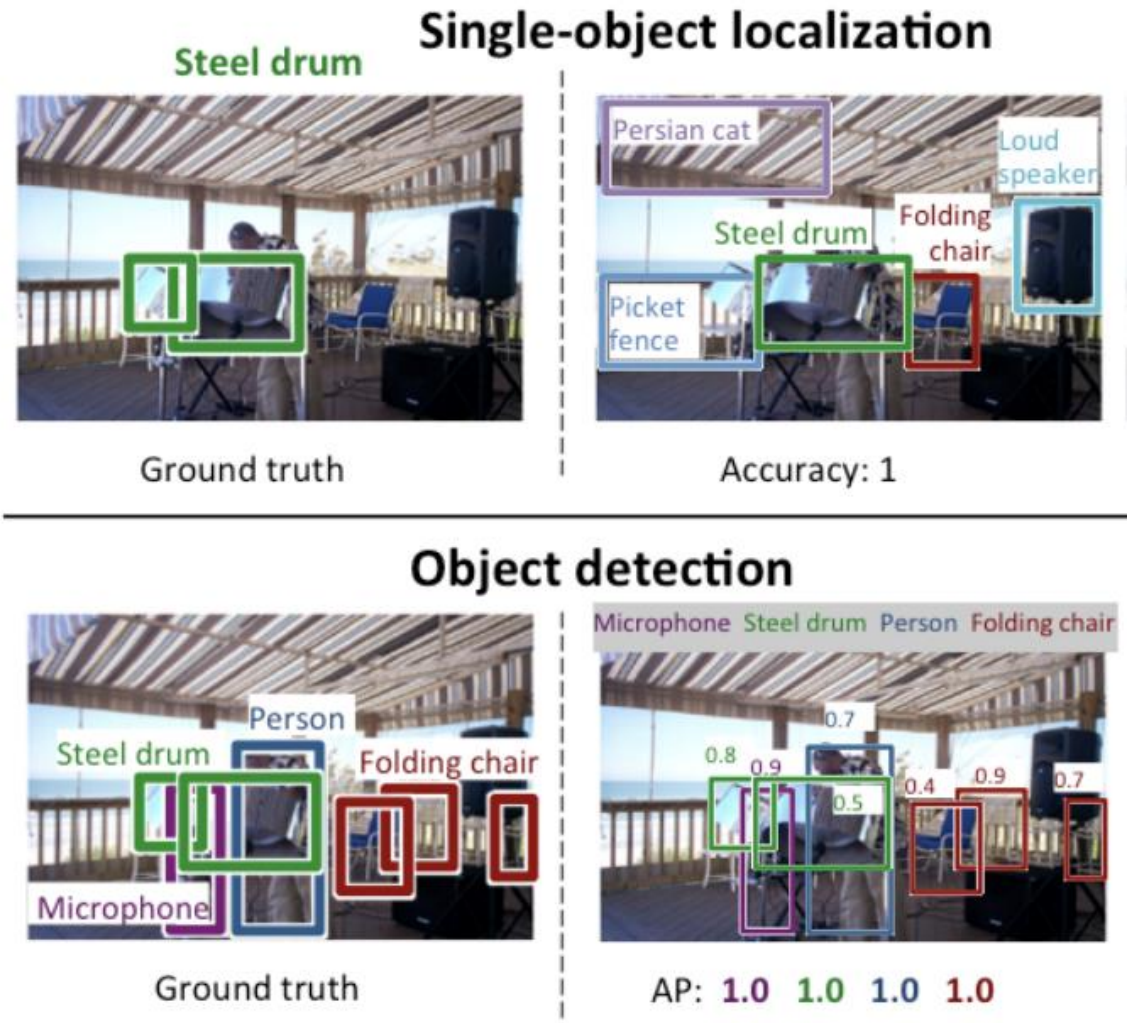
Ταξινόμηση εικόνας- Image classification: Οι αλγόριθμοι παράγουν μια λίστα κατηγοριών αντικειμένων που υπάρχουν στην εικόνα.

Εντοπισμός ενός αντικειμένου- Single-object localization: Οι αλγόριθμοι παράγουν μια λίστα κατηγοριών αντικειμένων που υπάρχουν στην εικόνα, μαζί με ένα πλαίσιο οριοθέτησης με άξονα που δείχνει τη θέση και την κλίμακα μιας παρουσίας κάθε κατηγορίας αντικειμένων.

Object detection-Ανίχνευση αντικειμένων: Οι αλγόριθμοι παράγουν μια λίστα κατηγοριών αντικειμένων που υπάρχουν στην εικόνα μαζί με ένα πλαίσιο οριοθέτησης με άξονα ευθυγραμμισμένο που δείχνει τη θέση και την κλίμακα κάθε παρουσίας κάθε κατηγορίας αντικειμένων[65].

Μπορούμε να δούμε ότι ο "Εντοπισμός ενός αντικειμένου" είναι μια απλούστερη έκδοση του ευρύτερα καθορισμένου "Τοπικοποίηση αντικειμένων", περιορίζοντας τις εργασίες εντοπισμού σε αντικείμενα ενός τύπου μέσα σε μια εικόνα, την οποία μπορούμε να υποθέσουμε ότι είναι ευκολότερη.

Ακολουθεί ένα παράδειγμα σύγκρισης εντοπισμού μεμονωμένου αντικειμένου και ανίχνευσης αντικειμένων, που έχει ληφθεί από τη δημοσίευση του ILSVRC[19][20].



Εικόνα 2-8:Σύγκριση μεταξύ "Εντοπισμού Μεμονωμένου Αντικειμένου" και "Ανίχνευσης Αντικειμένων"[18]

Σύγκριση μεταξύ "Εντοπισμού Μεμονωμένου Αντικειμένου" και "Ανίχνευσης Αντικειμένων". Σύμφωνα με το ImageNet Large Scale Visual Recognition Challenge.

Η απόδοση ενός μοντέλου για την ταξινόμηση εικόνας αξιολογείται χρησιμοποιώντας το μέσο σφάλμα ταξινόμησης στις προβλεπόμενες ετικέτες κλάσης. Η απόδοση ενός μοντέλου για εντοπισμό ενός αντικειμένου αξιολογείται χρησιμοποιώντας την απόσταση μεταξύ του αναμενόμενου και του προβλεπόμενου πλαισίου οριοθέτησης για την αναμενόμενη κατηγορία. Ενώ η απόδοση ενός μοντέλου για την αναγνώριση αντικειμένων αξιολογείται χρησιμοποιώντας την ακρίβεια και ανάκληση σε κάθε ένα από τα καλύτερα ταιριαστά πλαίσια οριοθέτησης για τα γνωστά αντικείμενα στην εικόνα.

[18] https://machinelearningmastery.com/object-recognition-with-deep-learning/, Εικόνα από: ImageNet Large Scale Visual Recognition Challenge.

2.6.2 Ψηφιακή επεξεργασία εικόνας

Η ψηφιακή επεξεργασία εικόνας είναι η χρήση του υπολογιστή για την επεξεργασία ψηφιακών εικόνων μέσω αλγορίθμων. Ως υποκατηγορία ή πεδίο της ψηφιακής επεξεργασίας σήματος, η ψηφιακή επεξεργασία εικόνας έχει πολλά πλεονεκτήματα σε σχέση με την αναλογική επεξεργασία εικόνας. Επιτρέπει μια πολύ ευρύτερη γκάμα αλγορίθμων για την εφαρμογή των δεδομένων εισαγωγής και μπορεί να αποφύγει προβλήματα όπως η συσσώρευση θορύβου και παραμόρφωσης κατά τη διάρκεια της επεξεργασίας. Δεδομένου ότι οι εικόνες ορίζονται σε δύο διαστάσεις (ίσως περισσότερες) η ψηφιακή επεξεργασία εικόνας μπορεί να μοντελοποιηθεί με τη μορφή πολυδιάστατων συστημάτων. Η παραγωγή και ανάπτυξη ψηφιακής επεξεργασίας εικόνας επηρεάζεται κυρίως από τρεις παράγοντες: πρώτον, την ανάπτυξη υπολογιστών. Δεύτερον, η ανάπτυξη των μαθηματικών (ειδικά η δημιουργία και η βελτίωση της διακριτής θεωρίας των μαθηματικών). Τρίτον, η ζήτηση για ένα ευρύ φάσμα εφαρμογών στο περιβάλλον, τη γεωργία, τον στρατό, τη βιομηχανία και την ιατρική επιστήμη έχει αυξηθεί.

Η ψηφιακή επεξεργασία εικόνας επιτρέπει τη χρήση πολύ πιο πολύπλοκων αλγορίθμων, και ως εκ τούτου, μπορεί να προσφέρει τόσο πιο εξελιγμένη απόδοση σε απλές εργασίες, όσο και την εφαρμογή μεθόδων που θα ήταν αδύνατες με αναλογικά μέσα.

Συγκεκριμένα, η επεξεργασία ψηφιακών εικόνων είναι μια συγκεκριμένη εφαρμογή και μια πρακτική τεχνολογία βασισμένη σε:

- Ταξινόμηση[66][67]
- Εξαγωγή χαρακτηριστικών
- Ανάλυση σήματος πολλαπλής κλίμακας
- Αναγνώριση μοτίβου
- Προβολή

Καθώς αναφερόμαστε σε πολλές έννοιες και εργασίες που μπορεί να κάνει η όραση του υπολογιστή ωστόσο αυτή η διάκριση δεν είναι απαραίτητα δύσκολη. Για παράδειγμα, η ταξινόμηση εικόνας είναι άμεσα συσχετισμένη με την ψηφιακή όραση, αλλά οι διαφορές μεταξύ εντοπισμού αντικειμένων και ανίχνευσης αντικειμένων μπορεί να προκαλέσουν σύγχυση, ειδικά όταν και οι τρεις εργασίες μπορούν να αναφέρονται εξίσου ως αναγνώριση αντικειμένων. Η ταξινόμηση εικόνας περιλαμβάνει την εκχώρηση μιας ετικέτας τάξης σε μια εικόνα, ενώ ο εντοπισμός αντικειμένων περιλαμβάνει την σχεδίαση ενός πλαισίου οριοθέτησης γύρω από ένα ή

περισσότερα αντικείμενα σε μια εικόνα. Η ανίχνευση αντικειμένων είναι πιο δύσκολη και συνδυάζει αυτές τις δύο εργασίες και σχεδιάζει ένα πλαίσιο οριοθέτησης γύρω από κάθε αντικείμενο που ενδιαφέρει την εικόνα και τους εκχωρεί μια ετικέτα κλάσης. Μαζί, όλα αυτά τα προβλήματα αναφέρονται ως αναγνώριση αντικειμένων.

Μέσω αυτής της εργασίας, θα ανακαλύψετε μια εισαγωγή για την αναγνώριση αντικειμένων και των προηγμένων μοντέλων βαθιάς μάθησης που έχουν σχεδιαστεί για την αντιμετώπισή του.

Οπότε θα υπάρχει διάκριση ότι:

Η αναγνώριση αντικειμένων αναφέρεται σε μια συλλογή σχετικών εργασιών για τον προσδιορισμό αντικειμένων σε ψηφιακές φωτογραφίες.

Τα περιφερειακά νευρωνικά δίκτυα με βάση την περιοχή, ή R-CNN, είναι μια οικογένεια τεχνικών για την αντιμετώπιση εργασιών εντοπισμού αντικειμένων και αναγνώρισης, σχεδιασμένα για απόδοση μοντέλου.

Το You Look Only Once, ή το YOLO, είναι μια δεύτερη οικογένεια τεχνικών αναγνώρισης αντικειμένων σχεδιασμένων για ταχύτητα και χρήση σε πραγματικό χρόνο

2.7 Πλατφόρμες Εφαρμογής Αναγνώρισης Αντικειμένων

2.7.1 Πλατφόρμα Tensorflow

Το Tensorflow[19] είναι μία end to end πλατφόρμα ανοιχτού κώδικα εκμάθησης μηχανών. Πρόκειται για μια βιβλιοθήκη μαθηματικών και χρησιμοποιείται επίσης για εφαρμογές μηχανικής μάθησης, όπως τα νευρωνικά δίκτυα. Χρησιμοποιείται τόσο για έρευνα όσο και για παραγωγή στην Google. Το TensorFlow αναπτύχθηκε από την ομάδα του Google Brain για εσωτερική χρήση της Google. Βγήκε με την Άδεια 2.0 της Apache στις 9 Νοεμβρίου 2015. Επίσης, δίνει τη δυνατότητα στους προγραμματιστές να δοκιμάσουν νέες βελτιστοποιήσεις και αλγόριθμους κατάρτισης. Το TensorFlow υποστηρίζει μια ποικιλία εφαρμογών, με επίκεντρο την εκπαίδευση και τα συμπεράσματα σε βαθιά νευρωνικά δίκτυα. Πολλές υπηρεσίες της Google χρησιμοποιούν την παραγωγή TensorFlow, την κυκλοφόρησαν ως έργο ανοιχτού κώδικα και έχουν χρησιμοποιηθεί ευρέως για την έρευνα μηχανικής μάθησης. Σε αυτή την εργασία, περιγράφουμε το μοντέλο ροής δεδομένων TensorFlow και επιδεικνύουμε την επιτακτική απόδοση που επιτυγχάνει η Tensor-Flow για διάφορες εφαρμογές πραγματικού κόσμου.

[19] https://www.usenix.org/system/files/conference/osdi16/osdi16-abadi.pdf

Το TensorFlow είναι ένα σύστημα δεύτερης γενιάς του Google Brain. Η έκδοση 1.0.0 κυκλοφόρησε στις 11 Φεβρουαρίου 2017. Ενώ η εφαρμογή αναφοράς λειτουργεί σε μεμονωμένες συσκευές, το TensorFlow μπορεί να λειτουργεί σε πολλαπλές CPU και GPU [68]

2.7.1.1 Προηγούμενο σύστημα: DistBelief

Το TensorFlow είναι ο διάδοχος του DistBelief, το οποίο είναι το κατανεμημένο σύστημα εκπαίδευσης νευρωνικών δικτύων που χρησιμοποιεί το Google από το 2011 . Το DistBelief χρησιμοποιεί την αρχιτεκτονική του διακομιστή παραμέτρων και εδώ βλέπουμε τους περιορισμούς του, άλλα συστήματα βασισμένα σε αυτήν την αρχιτεκτονική έχουν αντιμετωπίσει αυτούς τους περιορισμούς με άλλους τρόπους.

Παρόλο που το DistBelief επέτρεψε σε πολλά προϊόντα της Google να χρησιμοποιούν βαθιά νευρωνικά δίκτυα και αποτέλεσαν τη βάση πολλών ερευνητικών έργων μηχανικής μάθησης, σύντομα άρχισε να εμφανίζει περιορισμούς. Η διεπαφή δέσμης ενεργειών Python για τη σύνταξη προκαθορισμένων επιπέδων ήταν επαρκής για χρήστες με απλές απαιτήσεις, αλλά οι πιο προηγμένοι χρήστες μας αναζητούσαν τρία επιπλέον είδη ευελιξίας:

1. Καθορισμός νέων επιπέδων για αποτελεσματικότητα εφαρμόστηκαν τα επίπεδα DistBelief ως κλάσεις C ++. Η χρήση μιας ξεχωριστής, λιγότερο εξοικειωμένης γλώσσας προγραμματισμού για την υλοποίηση των επιπέδων αποτελεί εμπόδιο για τους ερευνητές μηχανικής μάθησης που επιδιώκουν να πειραματιστούν με νέες αρχιτεκτονικές , όπως οι δειγματοληπτικοί προγραμματιστές softmax.

2. Επαλήθευση των αλγορίθμων εκπαίδευσης. Πολλά νευρωνικά δίκτυα εκπαιδεύονται χρησιμοποιώντας στοχαστική κλίση (SGD), η οποία επαναλαμβάνει επαναληπτικά τις παραμέτρους του δικτύου μετακινώντας τις προς την κατεύθυνση που μειώνει μέγιστα την τιμή της συνάρτησης απώλειας. Διάφορες ανακοινώσεις προς τη SGD επιταχύνουν τη σύγκλιση με την αλλαγή του κανόνα ενημέρωσης. Οι ερευνητές συχνά θέλουν να πειραματιστούν με νέες μεθόδους βελτιστοποίησης, αλλά με το ίδιο τρόπο στο DistBelief συνεπάγεται την τροποποίηση της υλοποίησης του διακομιστή παραμέτρων.

3. Καθορισμός νέων αλγορίθμων κατάρτισης Οι εργαζόμενοι της DistBelief ακολουθούν ένα καθορισμένο πρότυπο εκτέλεσης: διαβάζουν μια παρτίδα δεδομένων εισόδου και τις τρέχουσες τιμές παραμέτρων, υπολογίζουν τη λειτουργία

απώλειας (ένα πέρασμα προς τα εμπρός μέσω του δικτύου), υπολογίζουν κλίσεις για κάθε μία από τις παραμέτρους. Αυτό το πρότυπο λειτουργεί για την εκπαίδευση απλών νευρωνικών δικτύων feed-forward, αλλά αποτυγχάνει για πιο εξελιγμένα μοντέλα, όπως επαναλαμβανόμενα νευρωνικά δίκτυα που περιέχουν βρόχους .

2.7.1.2 Αρχές Σχεδιασμού Tensorflow

Το TensorFlow σχεδιάστηκε για να είναι πολύ πιο ευέλικτο από το DistBelief, διατηρώντας ταυτόχρονα την ικανότητά του να ικανοποιεί τις απαιτήσεις του φόρτου εργασίας της μηχανής παραγωγής της Google. Το TensorFlow παρέχει μια απλή άντληση προγραμματισμού που βασίζεται στην ροή δεδομένων και επιτρέπει στους χρήστες να αναπτύξουν εφαρμογές σε κατανεμημένα συμπλέγματα, τοπικούς σταθμούς εργασίας, κινητές συσκευές και επιταχυντές που έχουν σχεδιαστεί ειδικά για το σκοπό αυτό. Μια διεπαφή δέσμης ενεργειών υψηλού επιπέδου περιβάλλει την κατασκευή γραφικών δεδομένων ροής δεδομένων και επιτρέπει στους χρήστες να πειραματιστούν με διαφορετικές αρχιτεκτονικές μοντέλων και αλγόριθμους βελτιστοποίησης χωρίς να τροποποιήσουν το κεντρικό σύστημα. Σε αυτήν την υποενότητα, επισημαίνουμε τις βασικές αρχές σχεδίασης του TensorFlow:

Τα διαγράμματα δεδομένων του πρωτεύοντος χειριστή. Τόσο το TensorFlow όσο και το DistBelief χρησιμοποιούν μια αναπαράσταση ροής δεδομένων για τα μοντέλα τους, αλλά η πιο εντυπωσιακή διαφορά είναι ότι ένα μοντέλο DistBelief περιλαμβάνει σχετικά λίγα σύνθετα "στρώματα", ενώ το αντίστοιχο μοντέλο TensorFlow αντιπροσωπεύει μεμονωμένους μαθηματικούς χειριστές (όπως πολλαπλασιασμό μήτρας, περιστροφή, κλπ.) ως κόμβοι στο γράφημα ροής δεδομένων. Αυτή η προσέγγιση διευκολύνει τους χρήστες να συνθέτουν νέα επίπεδα χρησιμοποιώντας μια διεπαφή δέσμης ενεργειών υψηλού επιπέδου. Πολλοί αλγόριθμοι βελτιστοποίησης απαιτούν από κάθε στρώση να έχει καθορισμένες κλίσεις και η δημιουργία δομών από απλούς χειριστές καθιστά εύκολη τη αυτόματη διαφοροποίηση αυτών των μοντέλων.

Αναστολή εκτέλεσης Μια τυπική εφαρμογή TensorFlow έχει δύο ξεχωριστές φάσεις: η πρώτη φάση ορίζει το πρόγραμμα (π.χ. ένα νευρωνικό δίκτυο που πρέπει να εκπαιδευτεί και οι κανόνες ενημέρωσης) ως ένα γραφικό συμβολικό γράφημα δεδομένων με σύμβολα κράτησης θέσης για τα δεδομένα εισόδου και τις μεταβλητές που αντιπροσωπεύουν την κατάσταση. και η δεύτερη φάση εκτελεί μια βελτιστοποιημένη έκδοση του προγράμματος στο σύνολο των διαθέσιμων συσκευών. Αναβάλλοντας την εκτέλεση έως ότου είναι διαθέσιμο όλο το πρόγραμμα, το TensorFlow μπορεί να βελτιστοποιήσει τη φάση εκτέλεσης χρησιμοποιώντας παγκόσμιες πληροφορίες σχετικά με τον υπολογισμό. Για παράδειγμα, το TensorFlow επιτυγχάνει υψηλή αξιοποίηση GPU χρησιμοποιώντας τη δομή εξάρτησης του γράφου για να εκδώσει μια ακολουθία πυρήνων στη GPU χωρίς να περιμένει ενδιάμεσα αποτελέσματα.

Το TensorFlow είναι διαθέσιμο σε πλατφόρμες υπολογιστών Linux, MacOS, Windows και κινητών υπολογιστών 64-bit συμπεριλαμβανομένων των Android και iOS. Η ευέλικτη αρχιτεκτονική της επιτρέπει την εύκολη ανάπτυξη υπολογισμών σε διάφορες πλατφόρμες (CPUs, GPUs, TPUs) και από επιτραπέζιους υπολογιστές σε συστοιχίες εξυπηρετητών, σε κινητές και edge συσκευές.

Το TensorFlow έχει παραχθεί σε Python (για την έκδοση 3.7 σε όλες τις πλατφόρμες) και C για τα API άλλες γλώσσες που χρησιμοποιήθηκαν είναι C++, Go, Java,JavaScript and Swift (early release).

Ένα σχηματικό γράφημα ροής δεδομένων TensorFlow για έναν αγωγό εκπαίδευσης, που περιέχει υπογράμματα για ανάγνωση δεδομένων εισόδου, προ επεξεργασία, εκπαίδευση και κατάσταση ελέγχου

2.7.1.3 Αρχιτεκτονική

Το TensorFlow Serving διευκολύνει την ανάπτυξη νέων αλγορίθμων και πειραμάτων, διατηρώντας παράλληλα την ίδια αρχιτεκτονική διακομιστών και API. Το TensorFlow Serving παρέχει εκτός πλαισίου ενσωμάτωση με τα μοντέλα TensorFlow, αλλά μπορεί εύκολα να επεκταθεί για να εξυπηρετήσει άλλους τύπους μοντέλων.

Servables

Τα Servables είναι η κεντρική ιδέα στο TensorFlow Serving. Τα Servables είναι τα υποκείμενα αντικείμενα που χρησιμοποιούν οι πελάτες για την εκτέλεση υπολογισμών (για παράδειγμα, μια αναζήτηση ή συμπέρασμα).

Το μέγεθος και η ευκρίνεια ενός Servable είναι ευέλικτα. Ένα ενιαίο Servable μπορεί να περιλαμβάνει οτιδήποτε από ένα μόνο κομμάτι ενός πίνακα αναζήτησης σε ένα μοντέλο σε μια πλειάδα μοντέλων συμπερασμάτων. Τα Servables μπορούν να είναι οποιουδήποτε τύπου και διεπαφής, επιτρέποντας ευελιξία και μελλοντικές βελτιώσεις όπως:

- streaming αποτελέσματα
- πειραματικά API
- ασύγχρονοι τρόποι λειτουργίας

Οι Servables δεν διαχειρίζονται τον δικό τους κύκλο ζωής. Τυπικά περιλαμβάνουν τα ακόλουθα:

1. ένα αρχείο TensorFlow SavedModelBundle (Tensorflow: Συνεδρία)
2. έναν πίνακα αναζήτησης για ενσωμάτωση ή αναζητήσεις λεξιλογίου

Τα Μοντέλα

Το TensorFlow Serving αντιπροσωπεύει ένα μοντέλο ως ένα ή περισσότερα servables που μπορούν να παρατηρηθούν. Ένα μοντέλο στο οποίο έχει εκπαιδευτεί μπορεί να περιλαμβάνει έναν ή περισσότερους αλγόριθμους (συμπεριλαμβανομένων των μαθηματικών βαρών) και πίνακες αναζήτησης ή ενσωμάτωσης. Μπορείτε να αναπαριστάτε ένα σύνθετο μοντέλο ως ένα από τα παρακάτω:

- πολλαπλές ανεξάρτητες όψεις
- ενιαίο σύνθετο

Τους Loaders

Οι Loaders διαχειρίζονται τον κύκλο ζωής του servable. Το API Loader επιτρέπει κοινή υποδομή ανεξάρτητα από συγκεκριμένους αλγόριθμους μάθησης, δεδομένα ή περιπτώσεις χρήσης προϊόντων. Συγκεκριμένα, οι Φορτωτές τυποποιούν τα API για τη φόρτωση και την εκφόρτωση ενός Servable.

Πηγές Sources

Οι πηγές είναι ενότητες plugin που βρίσκουν και παρέχουν servables. Κάθε πηγή παρέχει μηδενικές ή περισσότερες ορατές ροές. Για κάθε ροή servable, μια πηγή παρέχει ένα στιγμιότυπο Loader για κάθε έκδοση που καθιστά διαθέσιμη για φόρτωση. (Μια πηγή είναι στην πραγματικότητα αλυσοδεμένη μαζί με μηδέν ή περισσότερα SourceAdapters, και το τελευταίο στοιχείο της αλυσίδας εκπέμπει τους φορτωτές Loaders.)

Managers

Οι managers χειρίζονται τον πλήρη κύκλο ζωής των Servables, συμπεριλαμβανομένων:

- loading Servables
- εξυπηρετώντας το Servables
- εκφόρτωση Servables

Οι Managers λαμβάνουν Sources πηγές και παρακολουθούν όλες τις εκδόσεις. Ο Manager προσπαθεί να εκπληρώσει τα αιτήματα των Sources, αλλά μπορεί να αρνηθεί να φορτώσει μια έκδοση που έχει φιλοδοξεί, αν, για παράδειγμα, οι απαιτούμενοι πόροι δεν είναι διαθέσιμοι. Οι διαχειριστές μπορούν επίσης να αναβάλουν μια "εκφόρτωση". Για παράδειγμα, ένας διαχειριστής μπορεί να περιμένει να εκφορτωθεί μέχρι να ολοκληρωθεί η φόρτωση μια νεότερη έκδοση, με βάση μια πολιτική που εγγυάται ότι τουλάχιστον μία έκδοση είναι φορτωμένη ανά πάσα στιγμή.

Πυρήνας Core

Χρησιμοποιώντας το τυπικό APIS TensorFlow Serving, ο TensorFlow Serving Core διαχειρίζεται τις ακόλουθες πτυχές των servables:

- κύκλος ζωής
- μετρήσεις

Το TensorFlow Serving Core αντιμετωπίζει τα servables και τους loaders ως αδιαφανή αντικείμενα [9][10].

2.7.2 Πλατφόρμα OpenCV

Το OpenCV [20](ανοιχτού κώδικα) είναι μια βιβλιοθήκη λειτουργιών προγραμματισμού που στοχεύει κυρίως στην οπτικοποίηση σε πραγματικό χρόνο. Αρχικά αναπτύχθηκε από την Intel, στη συνέχεια υποστηρίχθηκε από το Willow Garage και στη συνέχεια το Itseez (το οποίο αργότερα αποκτήθηκε από την Intel). Η βιβλιοθήκη είναι πολλαπλών πλατφορμών και είναι δωρεάν για χρήση υπό την άδεια ανοικτού κώδικα BSD.Το OpenCV υποστηρίζει ορισμένα μοντέλα από βαθιά πλαίσια μάθησης όπως το TensorFlow.

Η βιβλιοθήκη έχει περισσότερους από 2500 βελτιστοποιημένους αλγορίθμους, οι οποίοι περιλαμβάνουν ένα ολοκληρωμένο σύνολο κλασικών και σύγχρονων υπολογιστικών όρων και αλγορίθμων μηχανικής μάθησης. Αυτοί οι αλγόριθμοι μπορούν να χρησιμοποιηθούν για την ανίχνευση και αναγνώριση προσώπων, τον εντοπισμό αντικειμένων, την ταξινόμηση των ανθρώπινων ενεργειών σε βίντεο, την παρακολούθηση κινήσεων κάμερας, την παρακολούθηση κινούμενων αντικειμένων, την εξαγωγή 3D μοντέλων αντικειμένων, την παραγωγή τρισδιάστατων σημείων από στερεοφωνικές κάμερες, εικόνα μιας ολόκληρης σκηνής, να βρείτε παρόμοιες εικόνες από μια βάση δεδομένων εικόνων, να αφαιρέσετε τα κόκκινα μάτια από τις εικόνες που τραβήξατε χρησιμοποιώντας το φλας, να παρακολουθήσετε τις κινήσεις των ματιών, να αναγνωρίσετε το τοπίο και να δημιουργήσετε δείκτες για να το επικαλύψετε με την επαυξημένη πραγματικότητα κλπ. Το OpenCV έχει περισσότερους από 47.000 χρήστες και εκτιμώμενο αριθμό λήψεων downloading άνω των 18 εκατομμυρίων. Η βιβλιοθήκη χρησιμοποιείται εκτενώς σε εταιρείες, ερευνητικές ομάδες και κυβερνητικούς φορείς [69].

Το OpenCV, το οποίο ξεκίνησε επίσημα το 1999, ήταν αρχικά μια πρωτοβουλία της Intel Research για την προώθηση εφαρμογών εντάσεως CPU, μέρος μιας σειράς έργων, όπως η ανίχνευση ακτίνων σε πραγματικό χρόνο και οι τρισδιάστατοι τοίχοι. Οι κύριοι συντελεστές του έργου περιλάμβαναν έναν αριθμό εμπειρογνωμόνων βελτιστοποίησης στη Intel Russia, καθώς

[20] https://en.wikipedia.org/wiki/OpenCV

και την ομάδα της Intel Performance Library. Στις πρώτες ημέρες του OpenCV, οι στόχοι του έργου περιγράφηκαν ως εξής:

Να προχωρήσει η έρευνα στον τομέα του οράματος παρέχοντας όχι μόνο ανοιχτό αλλά και βελτιστοποιημένο κώδικα για βασική υποδομή οράματος. Δεν υπάρχει πλέον επανεφεύρεση του τροχού.

Να διαδοθεί η γνώση παρέχοντας μια κοινή υποδομή την οποία θα μπορούσαν να αναπτύξουν οι προγραμματιστές, ώστε ο κώδικας να είναι πιο εύκολα αναγνώσιμος και μεταβιβάσιμος.

Οι εμπορικές εφαρμογές καθιστούν διαθέσιμο δωρεάν κώδικα βελτιστοποίησης απόδοσης - με άδεια που δεν απαιτεί τον κώδικα να είναι ανοιχτός ή ελεύθερος.

Η πρώτη alpha έκδοση του OpenCV κυκλοφόρησε στο κοινό στη διάσκεψη IEEE για το Computer Vision and Pattern Recognition το 2000 και πέντε beta κυκλοφόρησαν μεταξύ του 2001 και του 2005. Η πρώτη 1.0 έκδοση κυκλοφόρησε το 2006. Μια έκδοση 1.1 " "κυκλοφόρησε τον Οκτώβριο του 2008.

Το OpenCV είναι γραμμένο σε C ++ και το κύριο interface του είναι στο C ++, αλλά εξακολουθεί να διατηρεί μια λιγότερο εκτενή αλλά εκτεταμένη παλαιότερη διασύνδεση C. Υπάρχουν υλοποιήσεις σε Python, Java και MATLAB / OCTAVE. Το API για αυτές τις διεπαφές μπορεί να βρεθεί στην ηλεκτρονική τεκμηρίωση .

3 Περιβάλλον COSMOS

Το Mobile Edge Computing (MEC) είναι ένα βασικό συστατικό στοιχείο για την υλοποίηση των εφαρμογών του Internet of Things (IoT) στο έξυπνο περιβάλλον της πόλης. Το έργο του COSMOS[21] στόχευε στο σχεδιασμό και τη δοκιμή μιας εφαρμογής με δυνατότητα IoT για έξυπνες τουριστικές περιοχές αξιοποιώντας τις εγκαταστάσεις και τις λειτουργίες NFV / SDN του 5GINFIRE[71]. Οι επισκέπτες εκμεταλλεύονταν την εικόνα και τις δυνατότητες επεξεργασίας των φορητών συσκευών τους, εξοπλισμένες με κάμερες και διακομιστές στην άκρη του δικτύου, για να ανακτήσουν χρήσιμες πληροφορίες για κοντινά σημεία ενδιαφέροντος (PoIs).

Η διαδικασία περιλαμβάνει δύο στάδια: την αρχική αναγνώριση του PoI στην κινητή συσκευή και επιπλέον δεδομένα επεξεργασία από τους διακομιστές edge (edge servers) . Για να γίνει κατανοητό, το πλαίσιο της COSMOS [70], ενεργοποιήθηκε η δυναμική της επεξεργασίας της εικόνας από κινητές συσκευές σε edge clouds, κατά τα οποία τέτοιες υπηρεσίες (δηλ. με τη μορφή αλυσίδων VNF) μπορούν να αναπτυχθούν και να κλιμακωθούν. Θα χρησιμοποιηθεί ο αλγόριθμος πρόβλεψης κινητικότητα για την ακριβή εκτίμηση των εισερχόμενων αιτημάτων και τη διευκόλυνση του ακριβή σχεδιασμού των αλυσίδων VNF. Οι περιγραφές υπηρεσίας δικτύου (NSD) θα δημιουργηθούν και θα μεταφορτωθούν στο Open Source MANO (OSM) μέσω της πύλης 5GINFIRE για επεξεργασία εικόνας / βίντεο από MEC διακομιστές.

Ένα εργαλείο φόρτου εργασίας θα παρέχει μοντέλα που περιγράφουν τη δυναμική λειτουργία των VNF και υπολογίζει διάφορα σημεία λειτουργίας έναντι των διακυμάνσεων του φόρτου εργασίας. Ο πόρος, η περιγραφή αυτών των σημείων λειτουργίας θα συμπεριληφθεί στους περιγραφείς της υπηρεσίας δικτύου, δίνοντας τη δυνατότητα στη δυναμική ανάπτυξη της αλυσίδας VNF και την κλίμακα εισόδου / εξόδου στην υποδομή 5GINFIRE. Το πανεπιστήμιο Bristol 5G Testbed παρέχει την απαιτούμενη υποδομή 5GINFIRE, για την ικανοποίηση του πειραματισμού και επίδειξης της COSMOS. Ο γενικός και ολιστικός χαρακτήρας του, η προτεινόμενη προσέγγιση επιτρέπει την υιοθέτησή της και την προσαρμογή της σε ένα ευρύ φάσμα σεναρίων στο πλαίσιο του 5Gκαι την εποχή του IoT, όλοι αντιμετωπίζουν το ζήτημα της εξισορρόπησης φορτίου μεταξύ των συσκευών, βασισμένων σε άκρη και βάσει του cloud χρήση υπολογιστή.

Εκτός από τον αναμενόμενο επιστημονικό αντίκτυπο, η COSMOS φιλοδοξεί να προσφέρει διάφορους τύπους ανατροφοδότησης στην κοινοπραξία 5GINFIRE, ενισχύοντας τη λειτουργικότητά της, αυξάνοντας παράλληλα την προβολή της και μελλοντικές δυνατότητες.

[21] 5GINFIRE-D2-COSMOS-v1.0.pdf

3.1 Το έργο COSMOS

Το Mobile Edge Computing (MEC) είναι ένα βασικό στοιχείο για την υλοποίηση του Διαδικτύου των πραγμάτων (IoT) εφαρμογών στο πλαίσιο έξυπνης πόλης. Η αρχιτεκτονική MEC παρέχει τα πληθώρα πόρων του cloud computing στους χρήστες κινητών του δικτύου. Το έργο COSMOS στοχεύει στην ανάπτυξη μιας εφαρμογής με δυνατότητα IoT για αστικές τουριστικές περιοχές, τα μουσεία και τις πλατείες, μέσω της χρήσης των εγκαταστάσεων NFV / SDN που παρέχει η 5GINFIRE. Συγκεκριμένα, το Πανεπιστήμιο του Μπρίστολ 5G Testbed παρέχει την κατάλληλη MEC υποδομή στην πλατεία της Millennium στο κέντρο του Μπρίστολ. Οι επισκέπτες αυτού του πολυσύχναστου μέρους, θα χρησιμοποιήσουν συσκευές Raspberry Pi, εξοπλισμένες με κάμερες για λήψη στιγμιότυπων βίντεο μικρού μήκους από ένα σημείο ενδιαφέροντος (PoI) για να λάβουν χρήσιμες πληροφορίες για τον ιστότοπο μέσω της χρήσης της υπηρεσίας αναγνώρισης αντικειμένων. Για την αναγνώριση αντικειμένου, η εφαρμογή TensorFlow θα εγκατασταθεί τόσο σε κινητές συσκευές όσο και σε διακομιστές MEC.

Η προ επεξεργασία αρχικών δεδομένων θα πραγματοποιηθεί στη συσκευή Raspberry και εάν η αρχική επεξεργασία της έκβασης υποδεικνύει έναν πιθανό POI, τότε τα δεδομένα θα εκφορτωθούν μέσω Wi-Fi σε dedicated virtualized λειτουργίες δικτύου (VNFs) στο edge του δικτύου για περαιτέρω επεξεργασία.

Το πλεονέκτημα αυτής της επεξεργασίας διπλής βαθμίδας είναι διττό: (i) επέκταση της διάρκειας ζωής της μπαταρίας των φορητών συσκευών και (ii)ταχύτερη και ταυτόχρονη υποστήριξη πολλαπλών επισκεπτών με τη χρήση των διαθέσιμων επεξεργασίας ενέργειας στην άκρη του δικτύου.

Ο κύριος στόχος του COSMOS είναι να επιτρέψει τη δυναμική φόρτωση του φόρτου εργασίας των κινητών συσκευών σε σύννεφα άκρων (Edge Cloud) και ποσοτικοποιεί το χρονοδιάγραμμα κατά το οποίο τέτοιες υπηρεσίες (δηλ. μορφή αλυσίδων VNF) μπορούν να αναπτυχθούν και να κλιμακωθούν σε εξαιρετικά πυκνά περιβάλλοντα. Επιπροσθέτως, η COSMOS επικεντρώνεται στις μεθόδους πρόβλεψης της κινητικότητας των χρηστών για να παρέχει εγγυήσεις για την απόδοση την εφαρμογή για κινητά.

Από αυτήν την άποψη , δημιουργούνται περιγραφείς υπηρεσίας δικτύου (NSD) και μεταφορτώνονται στο Open Source MANO (OSM) μέσω της πύλης 5GINFIRE για επεξεργασία εικόνας / βίντεο εκφόρτωση σε διακομιστές MEC. Δυστυχώς, η έκδοση OSM 4 δεν παρέχει autoscaling χαρακτηριστικά και το OSM5 που παρέχει, αναμένεται. Για να ξεπεραστεί αυτό το εμπόδιο, σχεδιάστηκε και εφαρμόστηκε ένας μηχανισμός εξισορρόπησης φορτίου, βασισμένο σε ενεργά VM.

Στόχοι COSMOS	Αποτελέσματα
Μηχανισμός δυναμικής αποστολής/εκφόρτωσης υπολογισμών για μείωση επεξεργαστικής ισχύος των φορητών συσκευών	Οι συσκευές Raspberry PI 3 αποστέλλουν αυτόματα αιτήματα για περαιτέρω επεξεργασία στα VNF μέσω Wi-Fi.
Έλεγχος προφίλ φορτίου για τον υπολογισμό των πόρων που απαιτούνται για το VNF ώστε να γίνει αποτελεσματική τοποθέτηση VNF.	Η διαδικασία αναγνώρισης αναπτύσσεται προκειμένου να υπολογίζει συγκεκριμένα σημεία λειτουργίας, σε υπολογιστικούς πόρους, για κάθε VNF. Αυτή η διαδικασία είναι όσο το δυνατόν πιο γενική για να μην περιορίζεται σε συγκεκριμένη περίπτωση.. Επιπλέον, τα υπολογισμένα σημεία λειτουργίας διευκολύνει τη δυναμική κλιμάκωση της αλυσίδας VNF
Εκτίμηση κινητικότητας χρηστών	Η κινητικότητα των χρηστών μετριέται με αισθητήρες κίνησης και η εκτιμώμενη θέση λαμβάνεται υπόψη για το πότε θα αποφασιστεί η εκφόρτωση/αποστολή των δεδομένων.
Αξιολόγηση της Απόδοσης (μετρήσεις τροφοδοσίας και κλιμάκωση αυτής με χρονοδιαγράμματα, εντοπισμός bottlenecks) των VnF που θα αναπτυχθούν στο COSMOS	i)Εγκατάσταση του πειράματος ii) πειραματικά αποτελέσματα που δείχνουν την απόδοση από την επεξεργασία φόρτωσης iii) Η διάδοση των αποτελεσμάτων του έργου σε αξιόλογες διασκέψεις και επιστημονικά περιοδικά

Πίνακας 1-Στόχοι και αποτελέσματα COSMOS

Ένα πρόβλημα των εφαρμογών MEC και IoT είναι η υπολογιστική υπερφόρτωσης, δηλαδή η μεταφορά και η εκτέλεση εργασιών που απαιτούν υπολογιστικές ενέργειες από κινητές συσκευές στους MEC servers. Οι περισσότερες από τις προτεινόμενες μελέτες στη βιβλιογραφία σχετικά με την εκφόρτωση υπολογισμών, χρησιμοποιούν είτε μοντέλα θεωρίας ουράς αναμονής είτε σταθερά μοντέλα, για την επίλυση του πρόβλημα βελτιστοποίησης ανάλογα με την απαιτούμενη ενέργεια. Ωστόσο, η μεγάλη διαφορά μεταξύ

MEC και το περιβάλλον ενός σύννεφου είναι ότι οι πόροι των διακομιστών edge δεν είναι άφθονοι, και είναι δυναμικό workload profiling και ένας μηχανισμός κατανομής πόρων

είναι αναγκαία για να διασφαλιστεί η απόδοση και η αποτελεσματικότητα της επεξεργασίας των φορτίων. Επιπλέον, ο παράγοντας πρόβλεψης της ανθρώπινης κινητικότητας αποτελεί σημαντική παράμετρο στο πρόβλημα υπερφόρτωσης και έχουν ήδη προταθεί αρκετές λύσεις.

Πέρα από τις υπάρχουσες λύσεις, το έργο COSMOS αναπτύσσει μια καινοτόμο και ολιστική διαχείριση πόρων για εφαρμογές που επιτρέπουν τη χρήση του IoT. Πρώτον, μια γενική υπηρεσία προφίλ φόρτου εργασίας, με βάση σε μοντέλα state space, περιγράφει τη λειτουργία δυναμικών VNF και εντοπίζει διάφορα λειτουργικά σημεία μέσω της χρήσης ευέλικτων κριτηρίων QoS και QoE. Επιπλέον, το COSMOS η υπηρεσία εκτίμησης κινητικότητας διευκολύνει την πρόβλεψη και την αντιμετώπιση του μελλοντικού φόρτου εργασίας διακυμάνσεις. Τέλος, ο μηχανισμός εξισορρόπησης φορτίου της COSMOS αξιοποιεί με τον καλύτερο δυνατό τρόπο το κατανεμημένους πόρους στην αναπτυχθείσα εφαρμογή, λαμβάνοντας παράλληλα υπόψη την παραγωγή το προφίλ φόρτου εργασίας της εφαρμογής [6].

3.2 Θεωρητικό Υπόβαθρο

3.2.1 Απόφαση Εκφόρτωσης

Ένας από τους κύριους στόχους του πειράματος COSMOS είναι η ανάπτυξη ενός μηχανισμού εκφόρτωσης που βασίζεται σε σχετικές πληροφορίες και έχει ως στόχο να ικανοποιήσει τις απαιτήσεις QoS των χρηστών και του παρόχου υποδομής. Στο πλαίσιο της COSMOS, η απόφαση εκφόρτωσης έχει δύο φάσεις. Αρχικά, κάθε φορά που δημιουργείται ένα νέο αίτημα, η απόφαση εκφόρτωσης βασίζεται σε δύο παραμέτρους. α) τη μέτρηση της αντοχής του σήματος και β) τη θέση του χρήστη. Η θέση του χρήστη καθορίζεται με βάση τον αισθητήρα IMU και τη μεθοδολογία που περιγράφεται στην προηγούμενη υποενότητα. Η ισχύος του ασύρματου σήματος μετριέται χρησιμοποιώντας το λογισμικό Wavemon.

Η ένταση του σήματος είναι ένδειξη της ποιότητας του ρυθμού ασύρματης μετάδοσης. Σε πολυσύχναστες περιοχές, τα κανάλια της ασύρματης σύνδεση μοιράζονται μεταξύ πολλών χρηστών, πράγμα που σημαίνει ότι η παρεμβολή είναι υψηλή. Λαμβάνοντας υπόψη ότι η ισχύος μετάδοσης της κεραίας είναι οριοθετημένη, μόνο ένας μέγιστος αριθμός των χρηστών μπορεί να συνδεθεί. Επομένως, η ισχύος του ασύρματου σήματος επηρεάζει την απόφαση εκφόρτωσης , η οποία είναι αποδεκτή μόνο αν η ισχύος του σήματος είναι μεγαλύτερη από μια συγκεκριμένο όριο. Προκειμένου να διευκολυνθεί η αυτόματη εκφόρτωση, η τουριστική περιοχή χωρίζεται περαιτέρω όπου η εκφόρτωση επιτρέπεται βάσει της έντασης του σήματος.

Η κινητή συσκευή που βρίσκεται σε αυτό το "χάρτη εκφόρτωσης" με βάση τη μέτρηση του αισθητήρα IMU και τη μεθοδολογία που περιγράφεται παραπάνω. Εάν πληρούνται και οι

δύο περιορισμοί θέσης και σήματος, τότε η συσκευή αποσύρει το αίτημά της για περαιτέρω επεξεργασία. Κατά τη διάρκεια του δεύτερου σταδίου, η εξισορρόπηση φορτίου μηχανισμός που περιγράφεται παρακάτω, αποδέχεται ή απορρίπτει τα φορτία που έχουν εκφορτωθεί για την τελική επεξεργασία από τα VMs back-end.

3.2.2 Προφίλ πόρων

Το πρότυπο MEC παρέχει την απαραίτητη υποδομή πληροφορικής για την ανάπτυξη εφαρμογών με δυνατότητα IoT όπως στην περίπτωση χρήσης του COSMOS. Μια βασική διαφορά μεταξύ του MEC και toucloud αρχιτεκτονικές είναι ότι οι υπολογιστικοί πόροι του MEC είναι πεπερασμένοι και μια στατική κατανομή πόρων μηχανισμός συνήθως οδηγεί είτε σε υπέρβαση είτε σε χαμηλότερες προβλέψεις. Προς τούτο, ένας πόρος της εφαρμογής με δυνατότητα IoT διευκολύνει τη δυναμική κατανομή πόρων που χρησιμοποιήθηκε υποδομή MEC. Η απόδοση της εφαρμογής καθορίζεται από τις μετρήσεις QoS, π.χ. το χρόνο απόκρισης, τη διακίνηση ή την ενέργεια, και επηρεάζεται από πολλά δίκτυα και υπολογιστικές παραμέτρους, όπως η ποιότητα της ασύρματης σύνδεσης, η δυνατότητα παράλληλης την επεξεργασία των αιτημάτων και τον αριθμό των ταυτόχρονων χρηστών, η οποία είναι σημαντική για την περίπτωση COSMOS. Προκειμένου να απαντηθεί το ερώτημα εάν η εκφόρτωση ενός αιτήματος έχει νόημα ή όχι, πρέπει να έχουμε μια σαφή εικόνα για την απόδοση των τοπικών και απομακρυσμένων εκτελέσιμων αιτημάτων. Επιπλέον, πρέπει να γνωρίζουμε τον αντίκτυπο της απομακρυσμένης εκτέλεσης την υποδομή MEC. Έτσι, ένα δυναμικό προφίλ πόρων μπορεί να εξυπηρετήσει και τους δύο, τελικούς χρήστες και των παρόχων υποδομής.

Το COSMOS επικεντρώνεται στις έξυπνες τουριστικές εφαρμογές που παρέχονται σε πολυσύχναστους χώρους. Στο Μπρίστολ Millennium Square, οι επισκέπτες αξιοποιούν μια υπηρεσία αναγνώρισης εικόνων, με βάση toTensorFlow πρότυπο, για να ανακτηθούν πληροφορίες σχετικά με εκθέματα και γεγονότα που βρίσκονται κοντά. Η αναγνώριση ενός PoI από το TensorFlow είναι ένα υπολογιστικό έργο και η εκτέλεση του σε μια κινητή συσκευή απαιτεί μεγάλη ισχύ και χρόνο. Εκφορτώνει αυτές τις εργασίες σε ένα κέντρο δεδομένων MECεπιτρέπει την ταυτόχρονη επεξεργασία πολλαπλών αιτημάτων. Ωστόσο, ένας δυναμικός πόρος ο μηχανισμός κατανομής είναι απαραίτητος για τη μεγιστοποίηση των εξυπηρετημένων αιτημάτων και τη βέλτιστη αξιοποίηση της υποκείμενης υποδομής. Με αυτή την ικανότητα, ένας μηχανισμός δημιουργίας προφίλ πόρων πρέπει να ερμηνεύσει το ποσό και τις απαιτήσεις απόδοσης των φορτωμένων αιτημάτων σε κατανεμημένους πόρους από την πλευρά του MEC. Στο πλαίσιο αυτής της χωρητικότητας, υπολογίζουμε διάφορα λειτουργικά σημεία που αντιστοιχούν σε διαφορετικές συνθήκες λειτουργίας. Για παράδειγμα, ένα VM με 2 vCPUs και 4GB μνήμης RAM μπορούν να εξυπηρετήσουν 5 αιτήσεις ανά διάστημα

3.2.3 Πρόβλεψη φόρτου εργασίας

Ο αριθμός των επισκεπτών στην πλατεία Millennium Square ποικίλλει κατά τη διάρκεια της ημέρας. Κατά συνέπεια, ο όγκος των εκφορτωμένων αιτημάτων αλλάζει δυναμικά. Προκειμένου να επιτευχθεί βέλτιστη εξισορρόπηση φορτίου και την κατανομή των πόρων, απαιτείται μια μεθοδολογία πρόβλεψης του φόρτου εργασίας να υπολογίζει μελλοντικά αιτήματα στο backend στο επόμενο χρονικό διάστημα. Για το σκοπό αυτό, υιοθετήθηκε το Kalman Φιλτράρισμα , η οποία είναι μια πολύ γνωστή μεθοδολογία εκτίμησης για δυναμικά συστήματα.

3.2.4 Εξισορρόπηση φορτίου

Στο πλαίσιο αυτής της εργασίας, στοχεύουμε στην παροχή ενός ευφυούς δυναμικού βέλτιστου πόρου κατανομής. Ως εκ τούτου, ορίζουμε ένα συνολικό κόστος συστήματος που αναφέρεται στο κόστος της τοπολογίας που εφαρμόστηκε. Υποθέτουμε ότι μπορούμε να παράγουμε παραδείγματα m VMs με διαφορετικές προτιμήσεις. Αυτό το κόστος Cs είναι ανάλογα με το v που είναι ο αριθμός της vCPU που είναι αφιερωμένος σε ένα συγκεκριμένο VM. Με τους κατάλληλους μαθηματικούς τύπους έχουμε σαν αποτέλεσμα την επίλυση του προβλήματος βελτιστοποίησης σε κάθε χρονικό διάστημα Ως αποτέλεσμα, η

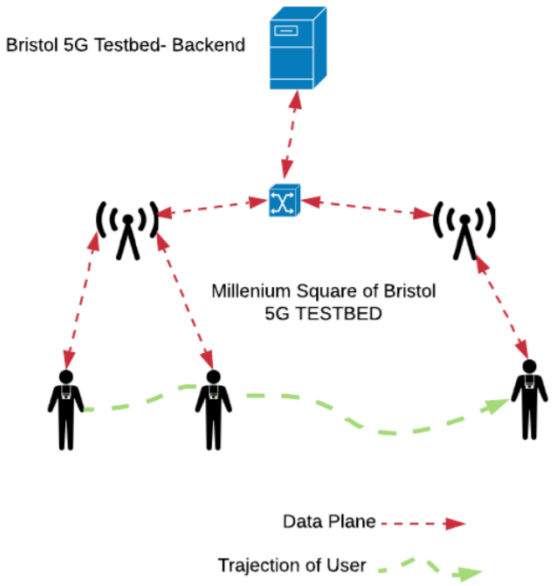
επίλυση αυτού του προβλήματος βελτιστοποίησης σε κάθε χρονικό διάστημα, το έργο COSMOS εξασφαλίζει ότι ο προβλεπόμενος φόρτος εργασίας θα εξυπηρετηθεί και οι ανάγκες των πόρων της VNF αλυσίδας θα ληφθούν υπόψη, ενώ θα αποφευχθεί η παροχή πόρων. Δυστυχώς, η προηγούμενη έκδοση OSM 4 δεν ενσωματώνει λειτουργίες αυτόματης κλιμάκωσης και η δυναμική απενεργοποίηση ή η εκκίνηση ενός VM στην υποδομή MEC του Μπρίστολ δεν είναι δυνατή. Επομένως, αν υποτεθεί ότι έχει πραγματοποιηθεί ήδη μια τοπολογία VM, εκτελούμε εξισορρόπηση φορτίου τα VM που θεωρούνται "λειτουργικά" για αυτό το διάστημα, εξαιρουμένων των VM που είναι που δεν έχει επιλεγεί από το πρόβλημα βελτιστοποίησης, να αναφέρεται στο εξής ως "ανενεργό".

3.3 Αρχιτεκτονική

3.3.1 Αρχιτεκτονική του COSMOS

Το σενάριο που αναφέρεται σε αυτό το έργο, κεφαλαιοποιείται στο Πανεπιστήμιο του Bristol 5G Testbed MEC υποδομή στο πλαίσιο της πλατείας Millennium στο κέντρο του Μπρίστολ, και στους χρήστες που κινούνται κοντά στην πλατεία. Οι επισκέπτες αυτού του γεμάτου χώρου θα χρησιμοποιήσουν το Raspberry Pi's, εξοπλισμένο με κάμερες για να τραβήξουν στιγμιότυπα ενός PoI και να λάβουν χρήσιμες πληροφορίες θέασης μέσω της υπηρεσίας αναγνώρισης αντικειμένων. Μέσω του της Google TensorFlow σύστημα, η υπηρεσία ταξινόμησης εικόνων (classify images) που επιλέχθηκε για την συγκεκριμένη περίπτωση είναι η "Inception v3" Deep Neural Network .

Οι συσκευές IoT είναι γενικά περιορισμένες όσον αφορά τη διαθεσιμότητα ενέργειας και την ισχύ επεξεργασίας. Ως εκ τούτου, για την ενίσχυση της εξοικονόμησης ενέργειας σύμφωνα με τις αρχές MEC για εφαρμογές με δυνατότητα IoT, χρήστες των κινητών συνδέονται μέσω πολλών σημείων ασύρματης πρόσβασης που βρίσκονται στην πλατεία Millennium Square και φορτώνουν τα αιτήματα επεξεργασίας δεδομένων σε ένα σύμπλεγμα διακομιστών Edge. Αυτή η τοποθέτηση επιτρέπει πρόσβαση χαμηλής καθυστέρησης στους διακομιστές, ικανή να εξυπηρετεί τα αιτήματα των χρηστών σε κατ' απαίτηση όπως απεικονίζεται στην εικόνα.



Εικόνα 3-1:Users Bristol[22]

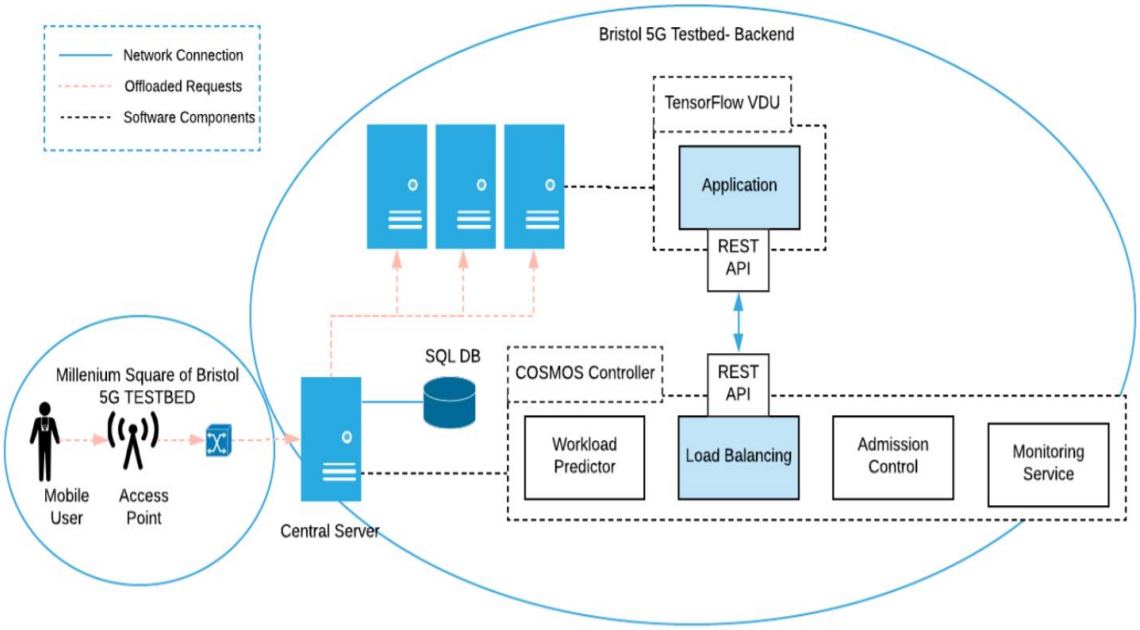
Σε αυτή την ενότητα περιγράφουμε αναλυτικά τον σχεδιασμό της αρχιτεκτονικής COSMOS. Το Μπρίστολ πανεπιστήμιο testbed παρέχει την έκδοση OSM 4 που λειτουργεί ως διαχείριση NFV και το οποίο είναι συνδεδεμένο με το OpenStack, το οποίο λειτουργεί ως Virtual Management Infrastructure Manager (VIM) που ελέγχει τις Μονάδες Εικονικής Ανάπτυξης (Virtual Deployment Units - VDU). Η αρχιτεκτονική του συστήματος του COSMOS ακολουθεί

[22] 5GINFIRE-D2-COSMOS-v1.0.pdf

ένα σχεδιασμό από την κορυφή προς τα κάτω, που σημαίνει ότι υπάρχει μια VDU, ονομαζόμενη κεντρικός ελεγκτής που υπαγορεύει τις αποφάσεις που απαιτούνται για την εξισορρόπηση φορτίου του εισερχόμενου φόρτου εργασίας, ενώ στο κάτω στρώμα είναι 3 VDU ονομαζόμενες Τα VDU TensorFlow με διαφορετικές προτιμήσεις που αναπτύχθηκαν από τον κόμβο OpenStack του Bristol. Η προτεινόμενη αρχιτεκτονική είναι γενικά εφαρμόσιμη σε υποδομές MEC ενός χώρου και μπορεί εύκολα να επεκταθεί στη συνεργασία Edge-to-Cloud ή Edge-to-Edge.

3.3.2 Ονομαζόμενες COSMOS

Η εκφόρτωση που δημιουργείται από τους χρήστες κινητών τηλεφώνων κατευθύνεται στον ελεγκτή COSMOSμέσω ενός ασύρματου σημείου πρόσβασης που βρίσκεται στην πλατεία Millennium. Εδώ βρίσκεται ο έλεγχος διαδικασία του μηχανισμού μας, όπως απεικονίζεται στην εικόνα 9. Για να συμπεριληφθεί αυτή η διαδικασία ελέγχου στο πλαίσιο μας, ο χρόνος είναι κβαντισμένος σε διακριτά χρονικά διαστήματα. Στην αρχή κάθε για κάθε διάστημα , ο ελεγκτής COSMOS παρακολουθεί τις εισερχόμενες αιτήσεις και επιλέγει την κατανομή στα τρία πιθανά VDU TensorFlow, για το εισερχόμενο φόρτο εργασίας για το επόμενο χρονικό διάστημα. Αυτή η διαδικασία ελέγχου, η οποία θα αναφέρεται στη συνέχεια ως Εξισορρόπηση Φορτίου, πραγματοποιείται με ένα on-line και ενεργό τρόπο, ενώ επικαλείται ένας εσωτερικός μηχανισμός πρόβλεψης, το Predictor Workload. Με βάση το φίλτρο Kalman, το τελευταίο παρέχει μια εκτίμηση του αναμενόμενου αριθμού αιτήσεων εντός του χρονικού διαστήματος. Επιπλέον, προκειμένου να διασφαλιστεί η ποιότητα QoE στους χρήστες, ο ελεγκτής της COSMOS, δέχεται μόνο έναν εκτιμώμενο αριθμό αιτημάτων και απορρίπτει εκείνους που το υπερβαίνουν. Επιπλέον, απορρίπτει τα αιτήματα από συσκευές με ισχύ σήματος κάτω από ένα σταθερό όριο. Αυτή η διαδικασία ελέγχου αναφέρεται ως έλεγχος εισαγωγής. Η βάση δεδομένων που περιέχει τα δεδομένα παρακολούθησης των αιτημάτων και οι επιδόσεις των VDUs λειτουργεί παράλληλα και βρίσκεται, επίσης, εδώ.



Εικόνα 3-2:Αρχιτεκτονική COSMOS[23]

3.3.3 VDUs TensorFlow

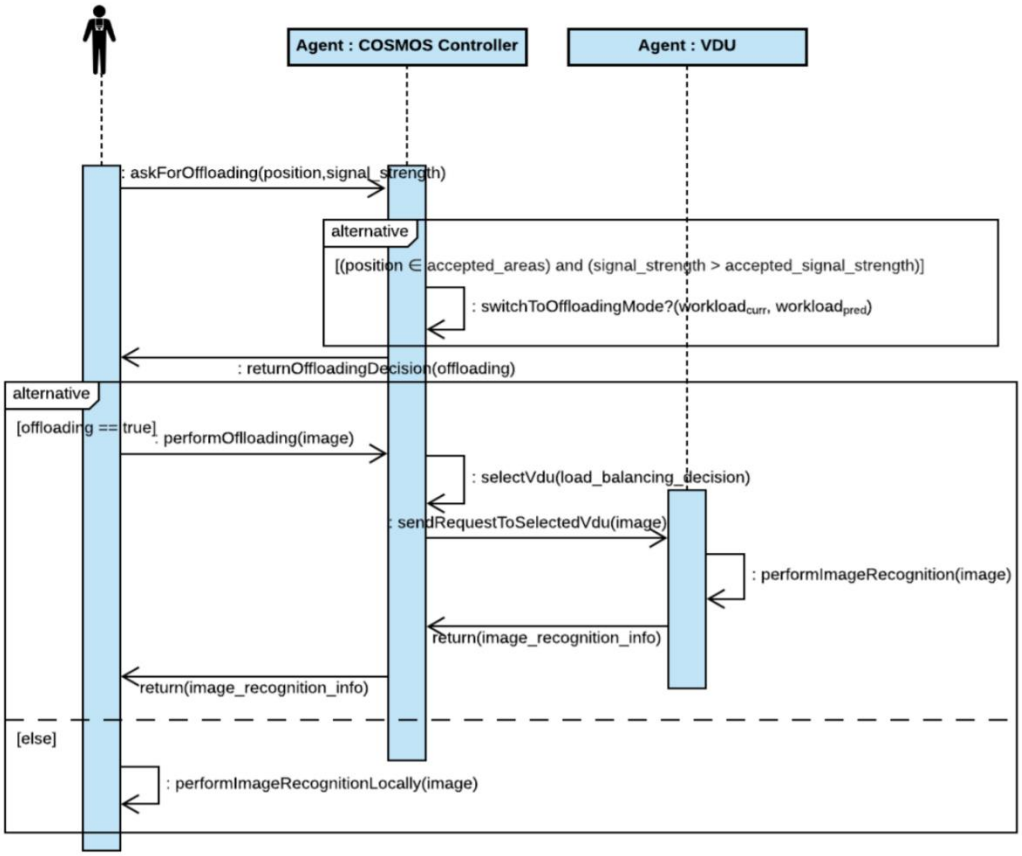
Οι υπηρεσίες αναγνώρισης αντικειμένων βάσει TensorFlow αναπτύσσονται ως VNF,επικοινωνώντας με τον ελεγκτή COSMOS μέσω ενός API REST που υπάρχει σε κάθε εφαρμογή VDU και αξιοποιώντας την εσωτερική σύνδεση που αναπτύχθηκε στο δίκτυο υπηρεσίας. Οι πόροι των VNF, δηλαδή των διαθέσιμων vCPU,η μνήμη και η αποθήκευση καθορίζονται από τις διάφορες προτιμήσεις.

3.3.4 Διαδικασία εκφόρτωσης

Σε αυτή την ενότητα, θα αναπτύξουμε περαιτέρω την αρχιτεκτονική σχεδιασμού του συστήματός μας σε ένα λογισμικό αντικειμενοστραφές βασισμένο σε component. Στο σενάριό , ο χρήστης βρίσκεται και κινείται κοντά στη πλατεία Millennium, δημιουργούν μια κίνηση αιτημάτων προς τις αναπτυσσόμενες μονάδες VDU στο MEC του Μπρίστολ. Η εικόνα 4 απεικονίζει λειτουργικότητα του ελεγκτή COSMOS, που δείχνει ρητά τη διαδρομή ενός φορτωμένου αιτήματος. Οι χρήστες στέλνουν ένα αρχικό αίτημα για να ενημερώσουν τον ελεγκτή COSMOS για τη θέση και την τρέχουσα ισχύος του σήματος.

[23] 5GINFIRE-D2-COSMOS-v1.0.pdf

Στη συνέχεια, ενημερώνονται εάν έχει γίνει δεκτή η αίτησή τους ανάλογα με ένα προκαθορισμένο όριο ισχύος του σήματος που σχετίζεται με τη θέση τους, με βάση τις μετρήσεις εκτός σύνδεσης της πλατείας Millennium Square. Επιπλέον, ο ελεγκτής COSMOS μπορεί να απορρίψει ένα συγκεκριμένο αίτημα εάν η συνολική ποσότητα φόρτου εργασίας για ένα δεδομένο χρονικό διάστημα είναι μεγαλύτερη από την προβλεπόμενη. Τέλος, οι χρήστες απαντούν με την εικόνα που θέλουν να εντοπίσουν και να λάβουν περισσότερες πληροφορίες σχετικά με, π.χ., τα αξιοθέατα στην πλατεία. Σύμφωνα με την απόφαση του COSMOS Controller στην αρχή κάθε διαστήματος, όπως περιγράφεται στην προηγούμενη ενότητα, το αίτημα ανακατευθύνεται στην κατάλληλη VDU TensorFlow για επεξεργασία. Στην περίπτωση απόρριψη αίτησης, οι χρήστες εκτελούν την επεξεργασία εικόνας τοπικά.



Εικόνα 3-3:Διάγραμμα Λειτουργικότητας Ελεγκτή COSMOS[24]

[24] 5GINFIRE-D2-COSMOS-v1.0.pdf

3.4 Εφαρμογή του COSMOS

Το πλαίσιο COSMOS αποτελείται από δύο βασικά στοιχεία. Ο ελεγκτής VNF υλοποιεί το μηχανισμό πληροφοριών της υπηρεσίας COSMOS και εξυπηρετεί τις λειτουργίες που έχουν αναπτυχθεί στο θεωρητικό υπόβαθρο του έργου, δηλαδή στην απόφαση εκφόρτωσης, στην κινητικότητα. Το δεύτερο στοιχείο είναι το VNF COSMOS-TensorFlow. Στο πλαίσιο του πειράματος COSMOS, το NSD δημιουργείται και φορτώνεται στον OSM μέσω της πύλης 5GinFIRE για επεξεργασία εικόνας και βίντεο και εκφόρτωση σε διακομιστές MEC. Εξάλλου, η COSMOS Ubuntu Image έχει μεταφορτωθεί στο OpenStack μέσω πύλης 5GinFIRE.

3.4.1 Λειτουργία Εικονικού Δικτύου (VNF)

Για τους σκοπούς του έργου COSMOS, το TensorFlow επεκτάθηκε προκειμένου να λειτουργήσει ως υπηρεσία και να λειτουργήσει ως VNF που επιτρέπει την ενσωμάτωση στο πλαίσιο OSM. Η βασική εικόνα, που χρησιμοποιήθηκε σε όλες τις πτυχές των VNF του πειράματος COSMOS, βασίστηκε στο Ubuntu Bionic18.04.2 LTS.Λαμβάνοντας υπόψη ότι η release έκδοση OSM 4.0 δεν υποστηρίζει κλιμάκωση, αποφασίσαμε να εφαρμόσουμε τρεις διαφορετικές επιλογές VDU (μικρή , μεσαία, μεγάλη).Στο τέλος της παραγράφου, παρουσιάζεται το αρχείο διαμόρφωσης YAML που εφαρμόζεται στον VNF COSMOS_big (VNFD),που χρησιμοποιείται στο σύνολο της υπηρεσίας δικτύου πειράματος COSMOS. Ομοίως, στις περιπτώσεις COSMOS_small και COSMOS_medium VNFDs, οι οποίες δεν παρουσιάζονται εδώ, η διαμόρφωση είναι παρόμοια. Ωστόσο, αυτές οι δύο εφαρμογές χρησιμοποιούν διαφορετικές επιλογές με χαμηλότερους πόρους που διατίθενται στην CPU, τη μνήμη και την αποθήκευση.

```
vnfd:vnfd-catalog:

    vnfd:
    -   id:: cosmos_big_vnfd
        name cosmos_big_vnfd
        short-name: cosmos_big_vnfd
        description: Ubuntu1804 with Tensorflow
        vendor: ICCS
        version: '1.5'

        logo: cosmos.png

        # Management interface
        mgmt-interface:
            cp: vnf-cp0

        vdu:

        -   id: cosmos_big_vnfd-VM
            name: cosmos_big_vnfd-VM
            description: cosmos_big_vnfd-VM
            count: 1
# Flavour of the VM to be instantiated for the VDU
vm-flavor:
    vcpu-count: 8
    memory-mb: 8192
      storage-gb: 20

# Image including the full path

      image: 'cosmos-tensorflow'
      cloud-init-file: cloud-init-1
      interface:

      -   name: eth0
          type: EXTERNAL
          virtual-interface:
              type: VIRTIO
          external-connection-point-ref: vnf-cp0
      -   name: eth1
          type: EXTERNAL
          virtual-interface:
              type: VIRTIO
          external-connection-point-ref: vnf-cp1

  connection-point:
  -   name: vnf-cp0
      type: VPORT
  -   name: vnf-cp1
      type: VPORT
```

```
vnf-configuration:
    juju:
        charm: identifier
   # day-2 configuration
    config-primitive:
    -   name: start-identifier
        parameter:
        -   name: host
            data-type: STRING
            mandatory: true
        -   name: port
            data-type: INTEGER
            mandatory: true
    #day-1 configuration
    initial-config-primitive:
    -   seq: '1'
        name: config
        parameter:
        -   name: ssh-hostname
            value: <rw_mgmt_ip>
        -   name: ssh-username
            value: cosmos
        -   name: ssh-password
            value: cosmos2019
```

Η περιγραφή του επάνω επιπέδου περιέχει βασικές πληροφορίες σχετικά με το VNFD όπως το όνομα και το αναγνωριστικό, καθώς και στοιχεία, π.χ. έκδοση κλπ. Το παραπάνω παράδειγμα VNFD περιγράφει τη διαμόρφωση που εφαρμόζεται στην ειδική εφαρμογή της VDU. Στην περίπτωση της COSMOS μεγάλο VNFD, μια εικονική μηχανή που διαθέτει 8GB μνήμης RAM, πρόσβαση σε 8 vCPU και 20GB αποθήκευσης πρέπει να παρουσιαστεί μέσα στην υπηρεσία nova-compute του OpenStack. Κάθε VNF έχει δύο σημεία σύνδεσης με έναν τύπο VPORT (vnf-cp0 και vnf-cp1). Το πρώτο σημείο σύνδεσης χρησιμοποιείται από το Juju Charms για τη διαμόρφωση της VDU, ενώ το δεύτερο το σημείο σύνδεσης επιτρέπει στο VDU να ανακτήσει τις φωτογραφίες που έχουν εκφορτώσει οι χρήστες, για να επικοινωνήσουν μαζί τους το διακομιστή Flask και, τέλος, επιστρέψει μετρήσεις όπως ο χρόνος που πέρασέ και η πρόβλεψη ακρίβειας, σε μορφή JSON.

3.4.2 Juju charms

3.4.1.2.1 Σκοπός

Το Juju[25] είναι ένα εργαλείο ενορχηστρωμένων υπηρεσιών cloud αναπτυγμένο σε Ubuntu, σε συνδυασμό με Ubuntu Server, Ubuntu OpenStack, MAAS (for bare-metal provisioning), και Landscape (για διαχείριση και παρακολούθηση συστημάτων)- Το Juju μας δίνει τη δυνατότητα να σχεδιάζουμε γρήγορα, να διαμορφώνουμε, να διαχειριζόμαστε, να διατηρήσουμε, να αναπτύξουμε και να κλιμακώσουμε υπηρεσίες cloud. Το Juju χρησιμοποιεί

[25] https://ubuntu.com/blog/12-questions-about-juju.

charms για να επιτύχει τις προαναφερθείσες λειτουργίες. Τα charms αποτελούνται από ένα σύνολο scripts για την ανάπτυξη και ένα λειτουργικό λογισμικό το οποίο μπορεί να εφαρμοστεί σε οποιαδήποτε εκτελέσιμη γλώσσα, π.χ. python, javaκλπ. Τα charms χρησιμοποιούνται μέσα στο OSM και πιο συγκεκριμένα μέσα στο αρχείο περιγραφής VNF για να καθοριστεί ολόκληρος ο κύκλος ζωής μιας εφαρμογής, συμπεριλαμβανομένων των ρυθμίσεων Day-0 Day-1 και Day-2. Η έκδοση OSM της πύλης 5GinFire υποστηρίζει τα charms που είναι υπεύθυνα μόνο για τη διαμόρφωση Day- 1και και Day-2.

3.4.3 Αναγνώστης - Charm

Για τους σκοπούς του πειράματος COSMOS, αναπτύχθηκε το identifier charm. Όπως φαίνεται στο εικόνα 6 και ελέγχει τη διαμόρφωση και τη διαχείριση του VNF. Στο πλαίσιο της δημιουργίας της υπηρεσίας ταυτοποίησης, δημιουργήθηκε μια δράση αναγνώρισης, η τελευταία περιέχει την περιγραφή υψηλού επιπέδου των δράσεων που υλοποιούνται στο εσωτερικό του identifier charm.

```
"start-identifier":
  "description": "Start the object detection server"
  "params":
    "host":
      "description": "The ip address of the host running this server"
      "type": "string"
      "default": "0.0.0.0"
    "port":
      "description": "The port the server listens to for incoming connections"
      "type": "string"
      "default": "5000"
  "required":
  - "host"
  - "port"
```

Εικόνα 3-4: Start identifier[26]

Απαιτούνται δύο παράμετροι για τη δράση εκκίνησης-αναγνωριστικού:

- Η θύρα -port
- Ο υπολογιστής-ip

Η παράμετρος κεντρικού υπολογιστή είναι η διεύθυνση IP του μηχανήματος που φιλοξενεί το διακομιστή. Αν δεν υπάρχει IP παρέχεται η διεύθυνση, η προεπιλεγμένη τιμή είναι

[26] 5GINFIRE-D2-COSMOS-v1.0.pdf

0.0.0.0. Η δεύτερη παράμετρος αφορά τη θύρα που ο διακομιστής ακούει τις εισερχόμενες συνδέσεις. Εάν δεν έχει οριστεί θύρα, η προεπιλεγμένη τιμή είναι για να χρησιμοποιήσετε τη θύρα 5000.

Η διαδικασία start-identifier απαιτεί την ενεργοποίηση του script app.py, έτσι πρέπει να εκτελέσουμε ως συστημένη υπηρεσία. Σε αυτό το σημείο, προχωράμε στη δημιουργία ενός συνημμένου αρχείου υπηρεσίας που περιλαμβάνει οδηγίες σχετικά με τον τρόπο εκτέλεσης του script app.py και στη συνέχεια ενεργοποιήστε το και ξεκινήστε χρησιμοποιώντας το systemd systemctl εργαλεία.

3.4.3.1 Αναγνωριστικό - διακομιστής εφαρμογής (Identifier - Application Server)

Ο διακομιστής εφαρμογής αναπτύχθηκε με τη χρήση του Flask. Το Flask είναι ένα python framework όπου εφαρμόζεται η προδιαγραφή WSGI. Αυτό το framework επιλέγεται λόγω του ελαφρού αποτυπώματος που επιτρέπει την ταχεία δημιουργία πρωτοτύπων και, επιπλέον, την αυξημένη δια λειτουργικότητα με το TensorFlow. Η κύρια χρήση του διακομιστή εφαρμογών είναι να εκθέσει ένα API έτσι ώστε οι καταναλωτές να μπορούν να δώσουν μια εικόνα και να πάρει τα αποτελέσματα συμπερασμάτων πίσω σε μια απάντηση. Αυτό επιτυγχάνεται εκθέτοντας ένα απλό API REST αποτελούμενο από ένα μόνο endpoint, δηλαδή την πρόβλεψη, υποστηρίζοντας ένα POST αίτημα ως τον τρόπο παροχής της εισερχόμενης εικόνας .

Ένας χειρισμός αιτήματος GET εφαρμόζεται μόνο στο πεδίο της επιστροφής ενός πιο ουσιαστικού μηνύματος στον χρήστη που έχει πρόσβαση στο API αντί ενός απλού 405 κωδικού http 405. Όλες οι απαντήσεις από τον διακομιστή εφαρμογών βρίσκονται σε JSON μορφή. Σε περιπτώσεις λανθασμένης απάντησης, ένα κλειδί "επιτυχίας" έχει την τιμή του να είναι ψευδής μαζί με ένα κλειδί "μηνύματος" του οποίου η τιμή περιέχει ένα ενδεικτικό αναγνωρίσιμο μήνυμα σφάλματος. Η απάντηση σε ένα επιτυχημένο αίτημα POST θα περιέχει ένα κλειδί "προβλέψεις" που διαθέτει το ίδιο σύνθετη δομή που περιέχει το όνομα του προσδιορισμένου αντικειμένου μαζί με την τιμή ακρίβειας της πρόβλεψης.

Ο διακομιστής εφαρμογής έχει ρυθμιστεί έτσι ώστε να μπορεί να υποστηρίζει λεπτομερή περιγραφή του TensorFlow μοντέλο και όλες τις διαδικασίες συμπερασμάτων. Δεδομένου ότι η δημιουργία προφίλ μπορεί να αποτελέσει σημαντική επιβάρυνση για την απόδοση του μοντέλου, μαζί με το μέγεθος της απάντησης, την είσοδο "profiling_on" POST το κλειδί αίτησης έχει οριστεί σε αληθές, ώστε να είναι δυνατή η άμεση ενεργοποίηση ή απενεργοποίηση.

Με την αρχικοποίηση του διακομιστή εφαρμογών, το γράφημα συμπερασμάτων φορτώνεται "απλά". Αυτό σημαίνει ότι όταν οι εξαρτήσεις φορτώνονται, το πραγματικό γράφημα συμπερασμάτων διαβάζεται όταν είναι πράγμα που απαιτείται για πρώτη φορά και,

μόλις διαβάσει, θα παραμείνει διαθέσιμο για το υπόλοιπο του κύκλου ζωής ενός συγκεκριμένου στιγμιότυπου της εφαρμογής

Κατά την ανάπτυξη του διακομιστή εφαρμογών, όλες οι αιτήσεις προς την εφαρμογή επιδόθηκαν μέσω του εσωτερικού διακομιστή Flask. Οι δυνατότητες αυτού του διακομιστή είναι περιορισμένες όσον αφορά την κλιμάκωση σε περιβάλλον παραγωγής, θα πρέπει να ενεργοποιηθεί η επεξεργασία πολλαπλών νημάτων (multi-threading processing).

3.4.4 Υπηρεσία δικτύου (Network Service -NS)

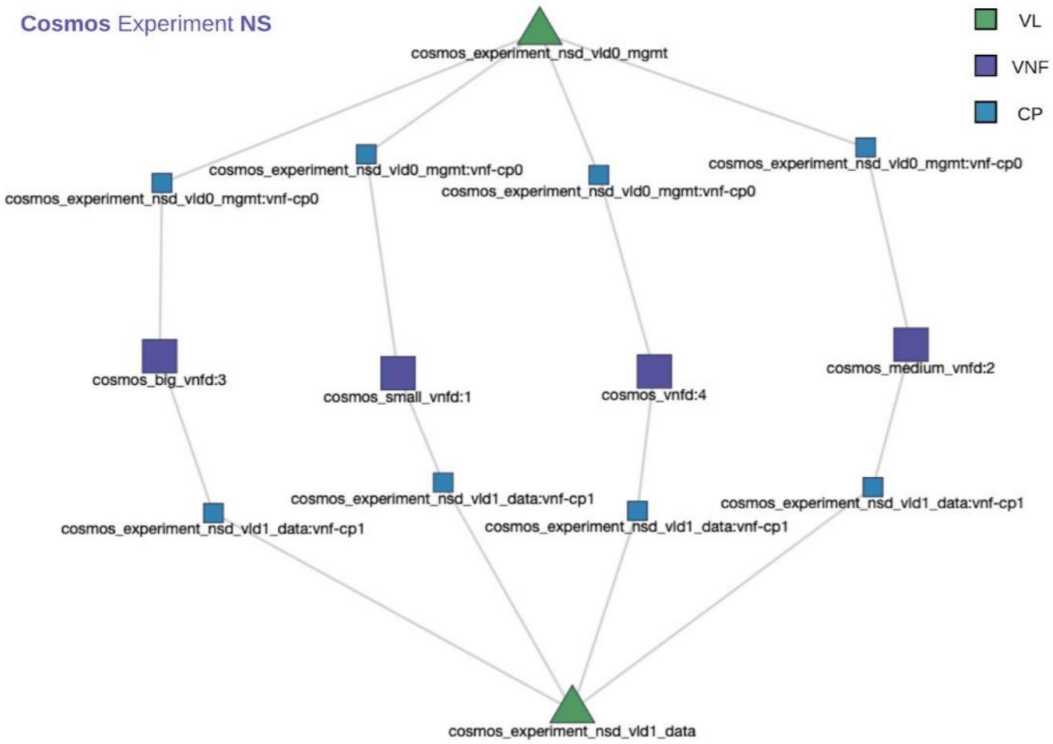
Το διάγραμμα της εικόνας 10 συνοψίζει τα βήματα της απομακρυσμένης επεξεργασίας ενός αίτημα εκφόρτωσης. Η υπηρεσία δικτύου COSMOS αποτελείται από τέσσερις VNF, δηλαδή τρεις VNF τρέχουν το TensorFlow και τον κύριο VNF ελεγκτή COSMOS.Ο ελεγκτής VNF υλοποιεί τους μηχανισμούς πληροφοριών της υπηρεσίας COSMOS και εξυπηρετεί τις λειτουργίες που έχουν αναπτύχθηκε στο θεωρητικό υπόβαθρο του έργου. Επιπλέον, η COSMOS NS περιλαμβάνει πρόσθετα τρία VDU, που αναπτύσσονται επίσης με το TensorFlow, και περιλαμβάνουν επιπλέον το εκπαιδευμένο μοντέλο καθώς και ένα διακομιστή εφαρμογών που χειρίζεται τις κλήσεις REST. Όλα τα VNFs του COSMOS βρίσκονται στο ίδιο δίκτυο και συνδέονται μεταξύ τους μέσω εικονικών συνδέσμων (virtual links). Τα API REST διευκολύνουν την ενδοεπικοινωνία των VNFs. Παρουσιάζονται οι ακόλουθες γραμμές το αρχείο YAML της COSMOS NSD:

```
nsd:nsd-catalog:
    nsd:
    -   id: cosmos_experiment_nsd
        name: cosmos_experiment_nsd
        short-name: cosmos_experiment_nsd
        description: Example NS for Cosmos experiment
        vendor: ICCS
        version: '1.5'

        logo: cosmos.png
        constituent-vnfd:
        -   member-vnf-index: 1
            vnfd-id-ref: cosmos_small_vnfd
        -   member-vnf-index: 2
            vnfd-id-ref: cosmos_medium_vnfd
        -   member-vnf-index: 3
            vnfd-id-ref: cosmos_big_vnfd
        -   member-vnf-index: 4
            vnfd-id-ref: cosmos_vnfd
vld:
# Networks for the VNFs
# management
-   id: cosmos_experiment_nsd_vld0_mgmt
    name: cosmos_experiment_nsd_vld0_mgmt
    short-name: cosmos_experiment_nsd_vld0_mgmt
    type: ELAN
    mgmt-network: 'true'
    vim-network-name: provider
```

```
        provider-network:
            segmentation_id: 1200
        vnfd-connection-point-ref:
        -   member-vnf-index-ref: 1
            vnfd-id-ref: cosmos_small_vnfd
            vnfd-connection-point-ref: vnf-cp0
        -   member-vnf-index-ref: 2
            vnfd-id-ref: cosmos_medium_vnfd
            vnfd-connection-point-ref: vnf-cp0
        -   member-vnf-index-ref: 3
            vnfd-id-ref: cosmos_big_vnfd
            vnfd-connection-point-ref: vnf-cp0
        -   member-vnf-index-ref: 4
            vnfd-id-ref: cosmos_vnfd
            vnfd-connection-point-ref: vnf-cp0
# data
-   id: cosmos_experiment_nsd_vld1_data
    name: cosmos_experiment_nsd_vld1_data
    short-name: cosmos_experiment_nsd_vld1_data
    type: ELAN
    vim-network-name: provider2
    provider-network:
        segmentation_id: 1201
    vnfd-connection-point-ref:
    -   member-vnf-index-ref: 1
        vnfd-id-ref: cosmos_small_vnfd
        vnfd-connection-point-ref: vnf-cp1
    -   member-vnf-index-ref: 2
        vnfd-id-ref: cosmos_medium_vnfd
        vnfd-connection-point-ref: vnf-cp1
    -   member-vnf-index-ref: 3
        vnfd-id-ref: cosmos_big_vnfd
        vnfd-connection-point-ref: vnf-cp1
    -   member-vnf-index-ref: 4
        vnfd-id-ref: cosmos_vnfd
        vnfd-connection-point-ref: vnf-cp1
```

Η εικόνα 11 απεικονίζει το γράφημα υπηρεσίας δικτύου COSMOS όπως απεικονίζεται στον πίνακα οργάνων OSM. Κάθε VNF αλληλοεπιδρά με δύο διαφορετικά επίπεδα, τη διαχείριση και τα δεδομένα. Η επικοινωνία μεταξύ τους διεξάγεται μέσω μιας εξειδικευμένης διεπαφής δικτύου που εκχωρείται σε αυτό το συγκεκριμένο επίπεδο. Αυτό επιτρέπει στο τελευταίο να έχει δικό του ξεχωριστό δίκτυο.



Εικόνα 3-5:Γράφημα υπηρεσίας δικτύου COSMOS[27]

Το επίπεδο του δικτύου διαχείρισης χρησιμοποιείται για να αλληλοεπιδράσει με το "VNF Configuration and Abstraction "και διευκολύνει την ανάπτυξη, κλιμάκωση και παροπλισμό σε πόρους, ενώ το επίπεδο του δικτύου δεδομένων χρησιμοποιείται για να διευκολύνει τις ανάγκες σε επίπεδο εφαρμογής Αυτό είναι το δίκτυο που χρησιμοποιείται για την επικοινωνία των καταναλωτών με τα VNF's και το API εφαρμογής.

3.5 Συμπεράσματα

Το παρόν κεφάλαιο παρέχει πληροφορίες σχετικά με το σχεδιασμό, την υλοποίηση, τον πειραματισμό, και τις προκλήσεις του σχεδίου COSMOS. Η προτεινόμενη αρχιτεκτονική και η εφαρμογή αναμένεται να είναι χρήσιμη για τους μελλοντικούς πειραματιστές και την κοινοπραξία 5GINFIRE, που ενδιαφέρονται για ενορχήστρωση υπολογιστικών συστημάτων και διαχείριση πόρων, υπολογισμοί της εκφόρτωσης και την εκτίμηση της κινητικότητας των χρηστών. Με βάση την εφαρμογή και την εκτέλεση, όλα οι στόχοι του έργου πληρούνται και συνοψίζονται ως εξής:

[27] 5GINFIRE-D2-COSMOS-v1.0.pdf

1. **Προφίλ πόρων:** Η απόδοση της υπηρεσίας αναγνώρισης εικόνων COSMOS, βασισμένη στο TensorFlow, διαμορφώνεται με την υιοθέτηση της δυναμικών συστημάτων που διευκολύνουν τον προσδιορισμό συγκεκριμένων σημείων λειτουργίας, τα οποία είναι αντιστοιχισμένα με τις διάφορες επιλογές των στιγμιότυπων VM.

2. **Πρόβλεψη κινητικότητας:** Υπολογίζεται η θέση και η τροχιά του χρήστη αξιοποιώντας τις δυνατότητες του αισθητήρα IMU και την προσέγγιση που βασίζεται τα βήματα στο τμήμα 2.1. Η εκτίμηση θέσης σε συνδυασμό με τη μέτρηση της ασύρματη ισχύος του σήματος επιτρέπει πιο ακριβή απόφαση εκφόρτωσης.

3. **Απόφαση εκφόρτωσης:** Η απόφαση εκφόρτωσης σχεδιάζεται με τέτοιο τρόπο ώστε να ικανοποιούνται οι απαιτήσεις QoS των χρηστών και αλληλοεπιδρά να αξιοποιήσει κατά βέλτιστο τρόπο τους υπολογιστικούς πόρους. Η αρχική απόφαση εκφόρτωσης βασίζεται στη θέση του χρήστη και της της ασύρματης σύνδεσης, ενώ η τελική απόφαση βασίζεται στον φόρτο εργασίας την πρόβλεψη και τη λύση εξισορρόπησης φορτίου.

4. **Εξισορρόπηση φορτίου:** Δεδομένου ότι η release έκδοση OSM 4 δεν παρέχει καμία δυνατότητα κλιμάκωσης, έχει σχεδιαστεί και εφαρμοστεί μηχανισμός εξισορρόπησης φορτίου . Ο μηχανισμός αυτός βασίζεται στο προσδιορισμό των πόρων και την πρόβλεψη φόρτου εργασίας με βάση το φίλτρο Kalman και τις ανακατευθύνσεις τα αιτήματα εκφόρτωσης στα ενεργά VDU για περαιτέρω επεξεργασία.

5. **Ενορχήστρωση OSM:** Ο ελεγκτής COSMOS VDU υλοποιεί όλα τις απαραίτητες λειτουργίες. Η υπηρεσία TensorFlow αναπτύσσεται σε τρεις VDU με διαφορετική ποσότητα των υπολογιστικών πόρων και οι βασικές VNFD δημιουργούνται. Οι απαραίτητες εικονικές συνδέσεις δημιουργούνται και περιλαμβάνονται στην NSD. Επιπλέον, JuJu Charms χρησιμοποιούνται για τον έλεγχο ολόκληρου του κύκλου ζωής των VDU

Βάσει των αρχικών αποτελεσμάτων, μπορούν να υπάρξουν σημαντικές βελτιώσεις για μελλοντική εργασία.

Πιο συγκεκριμένα:

1. Αξιοποιώντας τις δυνατότητες κλιμάκωσης των μελλοντικών εκδόσεων OSM και ειδικότερα την έκδοση 6, θα σχεδιάσουμε και θα αναπτύξουμε έναν αυτόματο μηχανισμό , ο οποίος θα ενεργοποιεί και απενεργοποιεί τα VDU.

2. Η μεθοδολογία πρόβλεψης του φόρτου εργασίας μπορεί να βελτιωθεί περαιτέρω, μολονότι προσεγγίζει ελαφριά μάθηση μηχανών[6][7][8].

4 Μεθοδολογία Εγκατάσταση TensorFlow & Εκπαίδευση Μοντέλου

Στα προηγούμενα κεφάλαια έγινε μια συνοπτική αναφορά στο Tensorflow και πως αυτό εμπλέκεται στις υποδομές του COSMOS, ωστόσο εδώ θα δοθεί μια πιο αναλυτική επεξήγηση τόσο της έννοιας του όσο και μία πρακτική σκοπιά για το πως μπορεί αυτό να εγκατασταθεί σε ποια περιβάλλοντα και πιο συγκεκριμένα πιο περιβάλλον χρησιμοποιήθηκε για το δικό μας πείραμα.

4.1 Τι είναι το Tensorflow

Το TensorFlow είναι μια ελεύθερη βιβλιοθήκη λογισμικού ανοιχτού κώδικα για τη ροή δεδομένων και τον προγραμματισμό σε μια σειρά εργασιών. Πρόκειται για μια βιβλιοθήκη μαθηματικών που χρησιμοποιείται για εφαρμογές μηχανικής μάθησης, όπως νευρωνικά δίκτυα. Χρησιμοποιείται τόσο για έρευνα όσο και για παραγωγή από την Google. Πιο συγκεκριμένα είναι μια end to end πλατφόρμα μηχανικής εκμάθησης και νευρωνικών δικτύων, ανοιχτού κώδικα.

Όπως αναφέρθηκε και στα προηγούμενα κεφάλαια η μηχανική μάθηση ή Machine learning είναι μια τεχνολογία υπολογισμού που εξελίσσεται συνεχώς τα τελευταία χρόνια και χρησιμοποιείται παντού γύρω μας. Η μηχανική μάθηση είναι η διαδικασία της εκπαίδευσης των υπολογιστών για να μάθουν πώς να αναλύουν δεδομένα και πως να παίρνουν τεκμηριωμένες αποφάσεις, χωρίς να προγραμματίζονται άμεσα γι’ αυτό.

Καθώς το TensorFlow είναι μια βιβλιοθήκη νευρωνικών δικτύων ανοιχτού κώδικα της Google, που αναπτύχθηκε από την ομάδα του Google Brain για πάρα πολλές χρήσεις. Ουσιαστικά το TensorFlow καταργεί την ανάγκη δημιουργίας ενός νευρικού δικτύου από την αρχή. Έτσι αφού υπάρχει ήδη η βάση, μπορεί αυτό μπορεί να εκπαιδευτεί με διαφορετικά δεδομένα κάθε στιγμή, να παραχθεί ένα νέο μοντέλο και να χρησιμοποιηθούν τα αποτελέσματα.

Η διαδικασία της εκπαίδευσης γίνεται με το image classification , με την παροχή εικόνων αναφοράς σε ένα νευρικό δίκτυο, το δίκτυο μπορεί να μάθει (μηχανικά) να προβλέπει αν μια εικόνα περιέχει παρόμοια αντικείμενα και να τα αναγνωρίζει. Η ίδια διαδικασία ελέγχθηκε και για το δικό μας πείραμα.[9]

4.2 Γιατί να χρησιμοποιηθεί το TensorFlow

Η μηχανική μάθηση δεν είναι εύκολη. Χρειάζεται καλή κατανόηση στατιστικών, μαθηματικών, προγραμματισμού και της γενικής επιστήμης των δεδομένων, καθώς όλα τα παραπάνω είναι απαραίτητα για να μάθουμε μια μηχανή να "σκέφτεται" και να αποφασίζει. Όμως το TensorFlow προσφέρει μαθήματα ακόμα και για αρχάριους. Τα επίσημα tutorials του TensorFlow σας καθοδηγούν βήμα προς βήμα για κάθε ρύθμιση και κάθε χρήση. Διαθέτει ένα ολοκληρωμένο, ευέλικτο οικοσύστημα εργαλείων, βιβλιοθηκών και πόρων που επιτρέπει στους ερευνητές και τους προγραμματιστές εύκολα να κατασκευάσουν και να αναπτύξουν εφαρμογές. Η μεγάλη κοινότητα πόρων και ανθρώπων δίνει την δυνατότητα και την πρόσβαση σε διαθέσιμο υλικό εκμάθησης.

Αυτά τα μαθήματα, παράλληλα με τη δωρεάν μηχανική μάθηση της Google, θα βοηθήσουν στην κατανόηση απόλυτα κάθε project. Το TensorFlow είναι ένα απίστευτα ισχυρό εργαλείο από την πιο μεγάλη εταιρεία του Διαδικτύου. Η απόφαση να το διαθέσει σαν open source, καθιστά την τεχνολογία προσιτή σε όλους .

4.3 API ανίχνευσης αντικειμένων TensorFlow

Ουσιαστικά αυτό που χρησιμοποιείται είναι το TensorFlow Object Detection API το οποίο αναπτύχθηκε ως ένα ευέλικτο σύστημα state-of-the-art μηχανικής εκμάθησης (ML) για τους υπολογιστές που όχι μόνο μπορεί να χρησιμοποιηθεί για τη βελτίωση των προϊόντων και των υπηρεσιών, αλλά και για να προωθηθεί πρόοδος στην ερευνητική κοινότητα. Η πρόοδος στην ερευνητική κοινότητα αναφέρεται κυρίως στην παρουσίαση της TF-Slim[28]: Μια βιβλιοθήκη υψηλού επιπέδου για τον καθορισμό σύνθετων μοντέλων στο TensorFlow, ένα ελαφρύ πακέτο για τον καθορισμό, την εκπαίδευση και την αξιολόγηση μοντέλων, καθώς και σημεία ελέγχου και ορισμούς μοντέλων για διάφορα ανταγωνιστικά δίκτυα στον τομέα της ταξινόμησης εικόνας. Η δημιουργία ακριβών μοντέλων ML ικανών να εντοπίσουν πολλαπλά αντικείμενα σε μία εικόνα παραμένει βασική πρόκληση στον τομέα αυτό και επενδύεται ένα σημαντικό χρονικό διάστημα εκπαίδευσης και πειραματισμού με αυτά τα συστήματα.

Λίγο αργότερα χρονικά από η ερευνητική ομάδα της Google σχετικά με την ανίχνευση αντικειμένων πέτυχε σημαντικά αποτελέσματα στο COCOAPI Challenge, πιο κάτω θα δούμε αναλυτικά σχετικά με αυτό το μοντέλο καθώς θα χρησιμοποιηθεί για την εκπαίδευση του custom μοντέλου. Το συγκεκριμένο API σύστημα έχει δημιουργήσει αποτελέσματα για διάφορες

[28] https://ai.googleblog.com/2016/08/tf-slim-high-level-library-to-define.html

ερευνητικές δημοσιεύσεις[293031] και έχει τεθεί σε λειτουργία σε προϊόντα Google όπως το NestCam, παρόμοια στοιχεία και οι ιδέες εμφανίζονται στην

Αναζήτηση Εικόνων το γνωστό προϊόν Image Search της εταιρείας , στην ανίχνευση αριθμού και ονόματος της οδού στο Street View.

Αυτό το σύστημα διατέθηκε στην ευρύτερη ερευνητική κοινότητα μέσω του API Object Detection TensorFlow. Αυτός ο κώδικας είναι ένα πλαίσιο ανοιχτού κώδικα που έχει δημιουργηθεί πάνω από το TensorFlow που διευκολύνει την κατασκευή, την εκπαίδευση και την ανάπτυξη μοντέλων ανίχνευσης αντικειμένων. Οι στόχοι στο σχεδιασμό αυτού του συστήματος ήταν να υποστηριχθούν τα υπερσύγχρονα μοντέλα, επιτρέποντας ταυτόχρονα την ταχεία εξερεύνηση και έρευνα.

4.3.1 Μεθοδολογία Εκπαίδευσης Μοντέλου

Παρακάτω ακολουθεί η μεθοδολογία εκπαίδευσης του custom μοντέλο, βασισμένη σε σύστημα Ubuntu, στο COCOAPI και σε Python εξασφαλίζοντας ότι υπάρχουν τα προαπαιτούμενα για την εγκατάσταση.

Διαδικαστικά η σειρά εγκαταστάσεων είναι:

- Εγκατάσταση του TensorFlow, είτε CPU ή GPU
- Εγκατάσταση των μοντέλων TensorFlow
- Εγκατάσταση εφαρμογής labelImg
- Εκπαίδευση μοντέλου
- Αποτελέσματα

Η οργάνωση και η σωστή μεθοδολογία βοηθούν στο να εξαλειφθούν οι λανθασμένες πρακτικές οπότε μία καλή πρακτική οργάνωσης είναι:

- Οργανώστε από την αρχή τα αρχεία χώρου εργασίας / εκπαίδευσης
- Τρόπος προετοιμασίας / σχολιασμού συνόλων δεδομένων εικόνας
- Δημιουργείστε εγγραφές tf από τέτοια σύνολα δεδομένων
- Εκπαιδεύστε το μοντέλο και να παρακολουθείτε την πρόοδό του
- Πώς να εξαγάγετε το μοντέλο που προκύπτει και να το χρησιμοποιήσετε για τον εντοπισμό αντικειμένων.

---

[29] Speed/accuracy trade-offs for modern convolutional object detectors, *Huang et al., CVPR 2017 (paper describing this framework)*

[30] Towards Accurate Multi-person Pose Estimation in the Wild, *Papandreou et al., CVPR 2017*

[31] YouTube-BoundingBoxes: A Large High-Precision Human-Annotated Data Set for Object Detection in Video, *Real et al., CVPR 2017 (see also our blog post)*

Για εγκατάσταση του σε CPU χρησιμοποιήθηκαν τα παρακάτω:

Το Tensorflow Object Detection API χρησιμοποιεί Protobufs για να παραμετροποιηθεί το μοντέλο και να εκπαιδευτούν οι παράμετροι. Πριν χρησιμοποιηθεί το Framework, οι βιβλιοθήκες του Protobuf θα πρέπει να εκτελεστούν. Αυτο εκτελείται από τον κατάλογο: tensorflow/models/research/ :

```
# From tensorflow/models/research/
protoc object_detection/protos/*.proto --python_out=.
```

Κατεβάζοντας τα κατάλληλα πακέτα και χρησιμοποιώντας έναν οδηγό εντολών μπορεί να γίνει η εγκατάσταση στα παραπάνω συστήματα. Επιπρόσθετα έχει δημιουργηθεί ένα ερευνητικό πρόγραμμα της Google για να βοηθήσει στη διάδοση της εκπαίδευσης και της έρευνας στον τομέα της μηχανικής μάθησης. Πρόκειται για ένα περιβάλλον σημειωματάριων λεγόμενα Jupyter που δεν απαιτεί εγκατάσταση αλλά εκτελείται στο μηχάνημα που βρίσκεται η υποδομή Tensorflow.

Η δομή των καταλόγων που θα περιέχουν τα αρχεία :

annotations: Αυτός ο φάκελος θα χρησιμοποιηθεί για την αποθήκευση όλων των αρχείων * .csv και των αντίστοιχων αρχείων TensorFlow *. records: αρχεία τα οποία περιέχουν τη λίστα σχολιασμών για τις εικόνες του συνόλου δεδομένων μας.

images: Αυτός ο φάκελος περιέχει ένα αντίγραφο όλων των εικόνων στο σύνολο δεδομένων μας, καθώς και τα αντίστοιχα αρχεία * .xml που παράγονται για κάθε μία, όταν χρησιμοποιείται το labelImg για τον σχολιασμό αντικειμένων.

images \ train: Αυτός ο φάκελος περιέχει ένα αντίγραφο όλων των εικόνων και τα αντίστοιχα αρχεία * .xml, τα οποία θα χρησιμοποιηθούν για την εκπαίδευση του μοντέλου μας.

images \ test: Αυτός ο φάκελος περιέχει ένα αντίγραφο όλων των εικόνων και τα αντίστοιχα αρχεία * .xml, τα οποία θα χρησιμοποιηθούν για τη δοκιμή του μοντέλου μας.

Pre-trained model: Αυτός ο φάκελος θα περιέχει το προ-εκπαιδευμένο μοντέλο της επιλογής μας, το οποίο θα χρησιμοποιηθεί ως σημείο εκκίνησης για την εκπαιδευτική μας εργασία.

training: Αυτός ο φάκελος θα περιέχει το αρχείο διαμόρφωσης αγωγού εκπαίδευσης * .config, καθώς και ένα αρχείο χάρτη ετικέτας *. pbtxt και όλα τα αρχεία που δημιουργήθηκαν κατά την εκπαίδευση του μοντέλου μας.

README.md: Πρόκειται για ένα προαιρετικό αρχείο που παρέχει μερικές γενικές πληροφορίες σχετικά με τις συνθήκες εκπαίδευσης του μοντέλου μας. Δεν χρησιμοποιείται από την TensorFlow με κανέναν τρόπο, αλλά γενικά βοηθά όταν έχετε μερικούς φακέλους προπόνησης ή / και επανεξετάζετε ένα εκπαιδευμένο μοντέλο μετά από κάποιο χρονικό διάστημα.

4.4 Μοντέλο COCO API

Για τη μεθοδολογία αναγνώρισης αντικειμένων θα χρησιμοποιηθεί είναι προ εκπαιδευμένο μοντέλο της κατηγορίας COCOApi

Το COCO (Common Objects in Context) είναι ένα σύστημα μεγάλης κλίμακας ανίχνευσης αντικειμένων, τμηματοποίησης και επεξήγησης τους. Το COCO έχει πολλά χαρακτηριστικά.

Τα σύνολα δεδομένων που σχετίζονται με την αναγνώριση αντικειμένων κατά προσέγγιση χωρίζονται σε τρεις ομάδες: αυτές που αφορούν κυρίως την ταξινόμηση αντικειμένων, ανίχνευση αντικειμένων και σημασιολογική «ετικετοποίηση» τους. Πιο κάτω να αναφερόμαστε στο κάθε ένα:

Ταξινόμηση εικόνας: Η λειτουργία της ταξινόμησης αντικειμένων απαιτεί δυαδικές ετικέτες που να δείχνουν εάν τα αντικείμενα δείχνουν μία εικόνα.

Ανίχνευση αντικειμένου: Η ανίχνευση ενός αντικειμένου συνεπάγεται δύο πράγματα δηλώνοντας ότι ένα αντικείμενο που ανήκει σε μια συγκεκριμένη κλάση είναι υπαρκτό και δεύτερο να εντοπιστεί στην εικόνα.

Σημασιολογική σήμανση αντικειμένου: Το καθήκον της σήμανσης σημασιολογικών αντικειμένων σε μια σκηνή απαιτεί κάθε εικονοστοιχείο ή εικόνα πρέπει να επισημαίνεται ότι ανήκει σε μια κατηγορία, όπως ουρανός, καρέκλα, δάπεδο, δρόμος κ.λπ.

Για όλα τα παραπάνω είναι εκπαιδευμένο το COCOAPI[32] συνεπώς η εκπαίδευση ενός custom μοντέλου βασίζεται σε ήδη προ εκπαιδευμένα μοντέλα όπως τα COCOAPI. Κάνουμε προσθήκη ενός τέτοιου μοντέλου στο σύστημα μας ώστε να προχωρήσει η εκπαίδευση βάση αυτού του μοντέλου.

[32] https://arxiv.org/pdf/1405.0312.pdf#page=12&zoom=100,65,100

Συγκεκριμένα για το πείραμα μας χρησιμοποιήθηκε η έκδοση ssd_mobilenet_v1_coco_11_06_2017. Εκτός από τους ορισμούς βασικών μοντέλων ανίχνευσης Tensorflow, αυτή η έκδοση περιλαμβάνει:

- Μια επιλογή εκπαιδευμένων μοντέλων ανίχνευσης, όπως:
- Single Shot Multibox Detector (SSD)[33] with MobileNet[34],
- SSD with Inception V2[35],
- Region-Based Fully Convolutional Networks (R-FCN) with Resnet 101,
- Faster RCNN[36] with Resnet 101,
- Faster RCNN with Inception Resnet v2
- Frozen weights (εκπαιδευμένα στο σύνολο δεδομένων COCO) για καθένα από τα παραπάνω μοντέλα που θα χρησιμοποιηθεί για σκοπούς εξαγωγής συμπερασμάτων.
- Ένα Jupyter notebook για εκτέλεση εμφάνισης αποτελεσμάτων

Επιπλέον προαπαιτούμενα για την σωστή εκπαίδευση του μοντέλου

Όταν τρέχουμε τοπικά, οι κατάλογοι tensorflow/models/research/ και slim θα πρέπει προσαρτηθούν στο PYTHONPATH. Αυτό επιτυγχάνεται τρέχοντας τα παρακάτω από τον κατάλογο tensorflow/models/research/:

```
# From tensorflow/models/research/
export PYTHONPATH=$PYTHONPATH:`pwd`:`pwd`/slim
```

4.5 Σχολιασμός αντικειμένου

Η διαδικασία συλλογής εικόνων είναι απλή, ωστόσο το MS COCO API έχει να παρουσιάσει ένα νέο σύνολο δεδομένων, μεγάλης κλίμακας που αντιμετωπίζει τρία βασικά ερευνητικά προβλήματα στην κατανόηση της σκηνής: ανίχνευση μη εικονικών προβολών (ή μη κανονικών προοπτικών[37]) αντικειμένων, συλλογιστική μεταξύ αντικειμένων και τον ακριβή 2D εντοπισμό αντικειμένων. Για πολλές κατηγορίες αντικειμένων, υπάρχει μια εικονική προβολή.

[33] https://arxiv.org/pdf/1512.02325.pdf

[34] https://arxiv.org/pdf/1704.04861.pdf

[35] https://arxiv.org/pdf/1512.00567v3.pdf

[36] https://arxiv.org/pdf/1803.06799.pdf

[37] S. Palmer, E. Rosch, and P. Chase, "Canonical perspective and the perception of objects," Attention and performance IX, vol. 1, p. 4, 1981.

Για παράδειγμα, όταν πραγματοποιείτε αναζήτηση εικόνων μέσω διαδικτύου για την κατηγορία αντικειμένων "ποδήλατο", ανακτήθηκε η κορυφαία κατάταξη παραδείγματα εμφανίζονται στο προφίλ, ανεμπόδιστα κοντά στο κέντρο μιας τακτοποιημένης φωτογραφίας. Θεωρούμε ότι αυτά το τρέχοντα συστήματα αναγνώρισης αποδίδουν αρκετά καλά σε εικονικές προβολές, αλλά αγωνίζονται να αναγνωρίσουν διαφορετικά αντικείμενα - στο παρασκήνιο, εν μέρει αποκρυμμένα, εν μέσω ανακατεμένα – πράγμα που αντικατοπτρίζει τη σύνθεση των πραγματικών καθημερινών σκηνών. Εμείς επαληθεύσαμε αυτό πειραματικά, αξιολογώντας καθημερινές σκηνές, μοντέλα που έχουν εκπαιδευτεί στα δεδομένα μας έχουν καλύτερη απόδοση από ό, τι εκείνα που εκπαιδεύτηκαν με προηγούμενα σύνολα δεδομένων.

Το Microsoft Common Objects in COntext (MS COCO) περιέχει 91 κοινές κατηγορίες αντικειμένων με 82 από αυτούς να έχουν περισσότερες από 5.000 παρουσίες με ετικέτα. Συνολικά το σύνολο δεδομένων έχει 2.500.000 επισημασμένες παρουσίες σε 328.000 εικόνες. Σε αντίθεση με το δημοφιλές σύνολο δεδομένων ImageNet[38], το COCO έχει λιγότερες κατηγορίες αλλά περισσότερες παρουσίες ανά κατηγορία. Αυτό μπορεί να βοηθήσει στη εκμάθηση στα μοντέλα αντικειμένων με δυνατότητα ακριβούς εντοπισμού 2D. Επιπλέον, μια κρίσιμη διάκριση μεταξύ του συνόλου δεδομένων μας και άλλα είναι ο αριθμός των παρουσιών με ετικέτα ανά εικόνα που μπορεί να βοηθήσει στην εκμάθηση πληροφοριών με βάση το περιεχόμενο.. Το MS COCO περιέχει πολύ περισσότερο αντικείμενα παρουσίες ανά εικόνα σε σύγκριση με το ImageNet (3,0)και PASCAL (2.3). Για πληρέστερη κατανόηση του τρόπου συλλογής και χαρακτηρισμού των εικόνων στο COCOAPI σύνολο δεδομένων είναι σημαντικό να δούμε αναλυτικά τις αναφορές του συγκεκριμένου κεφαλαίου.

Χρησιμοποιώντας ένα προ εκπαιδευμένο μοντέλο είναι πιο εύκολο να ιχνηλατηθεί οποιοδήποτε αντικείμενο καθώς τα προ εκπαιδευμένα μοντέλα έχουν βασιστεί στα μεγάλης κλίμακας σύνολα δεδομένων. Κατεβάζετε ή κάνετε μία μεγάλη συλλογή τις εικόνες που θέλετε να ιχνηλατηθούν, για το δικό μας πείραμα χρησιμοποιήθηκαν κινητά τηλέφωνα απαιτείται μία πληθώρα εικόνων όσο πιο πολλές με τόσο καλύτερη ακρίβεια θα εκπαιδευτεί το μοντέλο, περίπου 200 εικόνες χρησιμοποιήθηκαν σε αυτό εδώ το πείραμα.

Οι εικόνες πρέπει να αποτελούν ένα αντιπροσωπευτικό σύνολο του αντικειμένου που θα ανιχνευτεί, να είναι καλής ανάλυσης και να δίνουν ένα μεγάλο σύνολο δεδομένων

Σε συγκεκριμένο φάκελο images μέσα στο σύστημα εγκατάστασης αποθηκεύονται όλες οι εικόνες που θα χρησιμοποιηθούν για την αναγνώριση του αντικειμένου μας.

Σχετικά με τον σχολιασμό εικόνας (image Annonation) σε ένα μεγάλο σύνολο δεδομένων είναι μια σχετικά επίπονη διαδικασία, η πρώτη ενέργεια στο σχολιασμό του συνόλου

[38] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A Large-Scale Hierarchical Image Database,” in CVPR, 2009.

δεδομένων είναι να προσδιοριστούν ποιες κατηγορίες αντικειμένων υπάρχουν σε κάθε εικόνα Στο επόμενο στάδιο όλες οι εμφανίσεις των κατηγοριών αντικειμένων σε μια εικόνα επισημαίνονται, Στο προηγούμενο στάδιο επισημάνθηκε κάθε παρουσία μιας κατηγορίας, αλλά ενδέχεται να υπάρχουν πολλαπλές παρουσίες αντικειμένων. Το τελικό μας στάδιο είναι το επίπονο έργο της τμηματοποίησης της κάθε παρουσίας αντικειμένου, Εικ. 4.1

Εικόνα 4-1:Διαδικασία LambelImg

Σχετικά με το πείραμα η διαδικασία που γίνεται μ ένα γραφικό εργαλείο "σχολιασμού" ή μαρκάρισμα θα λέγαμε της εικόνας. Είναι γραμμένο σε Python και χρησιμοποιεί το Qt για τη γραφική διεπαφή του. Οι "σχολιασμοί" αποθηκεύονται ως αρχεία XML.

Ουσιαστικά επιλέγετε το αντικείμενο που θέλετε να αναγνωριστεί και σχολιάζετε το είδος του όπως φαίνεται στην εικόνα . Η ίδια διαδικασία ακολουθήθηκε και σ ένα custom μοντέλο. Μία εφαρμογή που μπορεί να χρησιμοποιηθεί είναι η labelImg[39] το οποίο αφορά περιβάλλοντα Ubuntu Linux με εντολές Python 3 .

Η σήμανση των εικόνων δεν απαιτεί ιδιαίτερη προσπάθεια, αν περιηγηθείτε στον κατάλογο images όπου υπάρχει το πλήθος των εικόνων και από όπου μπορείτε να ξεκινήσετε τον σχολιασμό των εικόνων.. Με τον ίδιο τρόπο κάνετε label όλες τις εικόνες ακόμη και αν δεν είναι ολόκληρες ή είναι διαφορετικές όψεις. Κάποιες φορές υπάρχουν και τα δύσκολα αντικείμενα, τι υποδηλώνει ότι το αντικείμενο έχει σημειωθεί ως "δύσκολο", για παράδειγμα, ένα αντικείμενο που είναι σαφώς ορατό αλλά δύσκολο να αναγνωριστεί χωρίς ουσιαστική χρήση του πλαισίου.

[39] https://github.com/tzutalin/labelImg

Μπορείτε να συμπεριλάβετε ή να αποκλείσετε δύσκολα αντικείμενα κατά τη διάρκεια της εκπαίδευσης.

Από την διαδικασία του σχολιασμού μίας εικόνας δημιουργείται για κάθε εικόνα αντίστοιχο xml αρχείο. Οπότε στον κατάλογο images υπάρχουν οι εικόνες που κατέβηκαν από το internet και τα αντίστοιχα xml αρχεία τους.

Στο παράδειγμα μας δημιουργήθηκαν δύο φάκελοι (directories) το train και το test. Το 10% των εικόνων που υπέστησαν επεξεργασία τα τοποθετούμε στον κατάλογο test και το άλλο 90% στον κατάλογο train, με την διαδικασία της αντιγραφής.

4.5.1 Διαχωρισμός των εικόνων

Μόλις ολοκληρώσετε τον σχολιασμό του συνόλου δεδομένων της εικόνας σας, είναι γενική διαδικασία να χρησιμοποιείτε μόνο μέρος αυτού για εκπαίδευση και το υπόλοιπο χρησιμοποιείται για σκοπούς αξιολόγησης (π.χ. όπως συζητήθηκε στην Αξιολόγηση του Μοντέλου (Προαιρετικό)).

Συνήθως, η αναλογία είναι 90% / 10%, δηλαδή 90% των εικόνων χρησιμοποιούνται για εκπαίδευση και το υπόλοιπο 10% διατηρείται για δοκιμή, αλλά μπορείτε να επιλέξετε ό, τι αναλογία ταιριάζει στις ανάγκες σας. Μόλις αποφασίσετε πώς θα χωρίσετε το σύνολο δεδομένων σας, αντιγράψτε όλες τις εικόνες εκπαίδευσης, μαζί με τα αντίστοιχα αρχεία * .xml και τοποθετήστε τις στο φάκελο training_demo \ images \ train. Παρομοίως, αντιγράψτε όλες τις δοκιμαστικές εικόνες, με τα αρχεία * .xml και επικολλήστε τις στο training_demo \ images \ test.

4.5.2 Δημιουργία Label Map

Το TensorFlow απαιτεί έναν χάρτη ετικετών, ο οποίος αντιστοιχεί συγκεκριμένα σε καθεμία από τις χρησιμοποιημένες ετικέτες σε ακέραιες τιμές. Αυτός ο χάρτης ετικετών χρησιμοποιείται τόσο από την εκπαίδευση όσο και από τις διαδικασίες ανίχνευσης.

Παρουσιάζεται ένα παράδειγμα χάρτη ετικέτας (π.χ. label_map.pbtxt), υποθέτοντας ότι το σύνολο δεδομένων μας περιέχει 2 ετικέτες, σκύλους και γάτες:

```
item {
  id: 1
  name: 'cat'
}

item {
  id: 2
  name: 'dog'
}
```

4.5.3 Μετατροπή των xml σε csv

Στο παράδειγμα μας θα χρησιμοποιηθεί ένα, σύνολο δεδομένων που έχει ήδη εκπαιδευτεί στην ανίχνευση Raccoon με το TensorFlow's Object Detection API. Από τον ιστότοπο https://GitHub.com/datitran/raccoon_dataset θα χρησιμοποιηθεί το xml_to_csv.py οπότε , οπότε στο ubuntu δημιουργείτε κατάλογο object-detection και μέσα σε αυτόν

τοποθετείται τον κατάλογο images και δημιουργείται αρχείο python με όνομα xml_to_csv.py , όπου αντιγράφετε τα περιεχόμενα του αντίστοιχου αρχείου από το dataset του raccoon.

Μόλις ολοκληρωθεί αυτή η διαδικασία το επόμενο στάδιο είναι η δημιουργία των TFRecords τα οποία είναι δυαδικής (binary) μορφής αρχεία του Tensorflow. Εάν εργάζεστε με μεγάλα σύνολα δεδομένων, η χρήση δυαδικής μορφής αρχείου για την αποθήκευση των δεδομένων σας μπορεί να έχει σημαντικό αντίκτυπο στην απόδοση κατά συνέπεια στον χρόνο εκπαίδευσης του μοντέλου. Τα δυαδικά δεδομένα καταλαμβάνουν λιγότερο χώρο στο δίσκο, παίρνουν λιγότερο χρόνο για αντιγραφή και μπορούν να διαβάσουν πολύ πιο αποτελεσματικά από το δίσκο.

4.5.4 Παραγωγή TFrecord αρχείων

Επιλέγοντας το αρχείο generate_tfrecord.py , με την ίδια διαδικασία δημιουργείστε στο object_detection αρχείο generate_tfrecord.py .

Η μοναδική αλλαγή που χρειάζεται να γίνει στο αρχείο είναι το σχόλιο αναγνώρισης των αντικειμένων , εδώ ήταν “raccoon” διαφοροποιείται πλέον με το όνομα αντικειμένου που θα αναγνωριστεί. Δημιουργούνται τα αρχεία τα train.record και test.record μετα τα csv.

Οπότε από τα αρχεία TFRecords και τα βήματα από labelImg στα TFRecords files που θα χρησιμοποιηθούν από το Object Detection API.

4.6 Εκπαίδευση του μοντέλου

Υπάρχουν δύο επιλογές. Μπορεί να χρησιμοποιηθεί ένα προ-εκπαιδευμένο μοντέλο και, στη συνέχεια, να γίνει μεταφερόμενη μάθηση για να αναγνωριστεί ένα νέο αντικείμενο ή θα μπορούσε να αναγνωρίσουμε νέα αντικείμενα εξ ολοκλήρου. Το όφελος της εκμάθησης από μεταφορά είναι ότι η εκπαίδευση μπορεί να είναι πολύ πιο γρήγορη και τα απαιτούμενα δεδομένα που μπορεί να χρειαστείτε είναι πολύ λιγότερα. Γι’ αυτόν τον λόγο, πρόκειται να γίνει μεταφερόμενη μάθηση. Για τους σκοπούς αυτού του πειράματος δεν θα δημιουργηθεί μια εκπαιδευτική διαδικασία από το μηδέν, αλλά θα εξεταστεί πώς να επαναχρησιμοποιηθεί ένα από τα προ-εκπαιδευμένα μοντέλα που παρέχονται από την TensorFlow.

Η διαδικασία εκπαίδευσης μπορεί να κρατήσει από 1 έως μερικές ώρες εξαρτάται αν υπάρχει GPU ή CPU. Για το παράδειγμα μας οι προδιαγραφές του μηχανήματος που έγινε το training είναι με CPU οπότε θα πάρει μερικές ώρες. Για να γίνει αυτό, χρειάζονται εικόνες, που ταιριάζουν με τα TFRecords για τα δεδομένα εκπαίδευσης και δοκιμών, και στη συνέχεια πρέπει να ρυθμίσουμε το μοντέλου, κατόπιν μπορεί να εκπαιδευτεί. Αυτό σημαίνει ότι πρέπει να ρυθμιστεί ένα αρχείο με παραμέτρους (configuration file).

Το TensorFlow διαθέτει αρκετά προ-εκπαιδευμένα μοντέλα με διαθέσιμα αρχεία ελέγχου, μαζί με αρχεία ρυθμίσεων. Το μοντέλο που θα χρησιμοποιηθεί στα παραδείγματά μας είναι το μοντέλο ssd_mobilenet_v1_pets.config , δεδομένου ότι παρέχει μια σχετικά καλή αντιστάθμιση μεταξύ απόδοσης και ταχύτητας, ωστόσο υπάρχουν ορισμένα άλλα μοντέλα που μπορείτε να χρησιμοποιήσετε, τα οποία παρατίθενται repository του TensorFlow. Περισσότερες πληροφορίες σχετικά με την απόδοση ανίχνευσης, καθώς και τους χρόνους αναφοράς εκτέλεσης, για καθένα από τα διαθέσιμα προ-εκπαιδευμένα μοντέλα μπορείτε να βρείτε εδώ[40].

Στο αρχείο ρυθμίσεων, πρέπει να τροποποιηθούν όλα τα PATH_TO_BE_CONFIGURED και να αλλαχθούν. Μπορεί επίσης να θέλετε να τροποποιήσετε το μέγεθος της παρτίδας. Προς το παρόν, έχει οριστεί σε 24 στο αρχείο διαμόρφωσής μας. Άλλα μοντέλα μπορεί να έχουν διαφορετικά μεγέθη παρτίδων. Αν εμφανίζεται σφάλμα μνήμης, μπορείτε να προσπαθήσετε να μειώσετε το μέγεθος της παρτίδας για να ταιριάξει το μοντέλο στη VRAM που έχετε. Τέλος, πρέπει επίσης να αλλάξετε τα checkpoint name/path, num_classes to 1, num_examples to 12, and label_map_path: "training/object-detect.pbtxt".

Η εκκίνηση της εκπαίδευσης του μοντέλου είναι το τελευταίο βήμα, η βασική εντολή της έναρξης:

Ξεκινάει η διαδικασία της εκπαίδευσης. Τα βήματά ξεκινούν από το 1 και η απώλεια θα είναι πολύ υψηλότερη. Ανάλογα με τη CPU και πόσα εκπαιδευτικά δεδομένα εισόδου έχουν δοθεί στο μοντέλο, αυτή η διαδικασία θα διαφέρει χρονικά. Στο δικό μας παράδειγμα η εκπαιδευτική διαδικασία κράτησε πάνω από 20 ώρες. Αν έχετε πολλά εκπαιδευτικά δεδομένα, ίσως χρειαστεί πολύ περισσότερο. Θέλετε η απώλεια να είναι στο ~ 1 κατά μέσο όρο (ή χαμηλότερη). Δεν σταμάτησε η εκπαίδευση μέχρι περίπου στο 1. Μπορείτε να ελέγξετε το μοντέλο μέσω του TensorBoard. Ο κατάλογός μοντέλων / object_detection / training θα έχει νέα αρχεία συμβάντων που μπορούν να προβληθούν μέσω του TensorBoard.

Οι χρόνοι εκπαίδευσης μπορούν να επηρεαστούν από διάφορους παράγοντες όπως:

- Η υπολογιστική ισχύος του hardware (είτε CPU είτε GPU): Προφανώς, όσο πιο ισχυρός είναι ο υπολογιστής , τόσο πιο γρήγορη είναι η διαδικασία εκπαίδευσης.
- Εξαρτάται αν χρησιμοποιείτε την παραλλαγή TensorFlow CPU ή GPU: Σε γενικές γραμμές, ακόμη και σε σύγκριση με τις καλύτερες CPU, σχεδόν οποιαδήποτε κάρτα γραφικών GPU θα αποφέρει πολύ πιο γρήγορες ταχύτητες εκπαίδευσης και ανίχνευσης. Ένα παράδειγμα εκτέλεσης είναι εκτελέστηκε το TensorFlow στο Intel i7-5930k (6/12 πυρήνες @ 4GHz, 32 GB RAM) και λήφθηκαν χρόνοι βήμα περίπου 12 δευτερόλεπτα / βήμα, μετά τον

40 https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/detection_model_zoo.md#coco-trained-models-coco-models

οποίο εγκαταστάθηκε σε TensorFlow GPU και εκπαιδεύοντας το ίδιο μοντέλο - χρησιμοποιώντας το ίδιο σύνολο δεδομένων και αρχεία ρυθμίσεων - σε ένα EVGA GTX-770 (1536 CUDA-core @ 1GHz, 2GB VRAM) είχαν μειωθεί στα 0,9 δευτερόλεπτα / βήμα . Μια 12πλάσια αύξηση της ταχύτητας, χρησιμοποιώντας μια κάρτα γραφικών “low / mid-end”, σε σύγκριση με μια CPU “mid / high-end”.

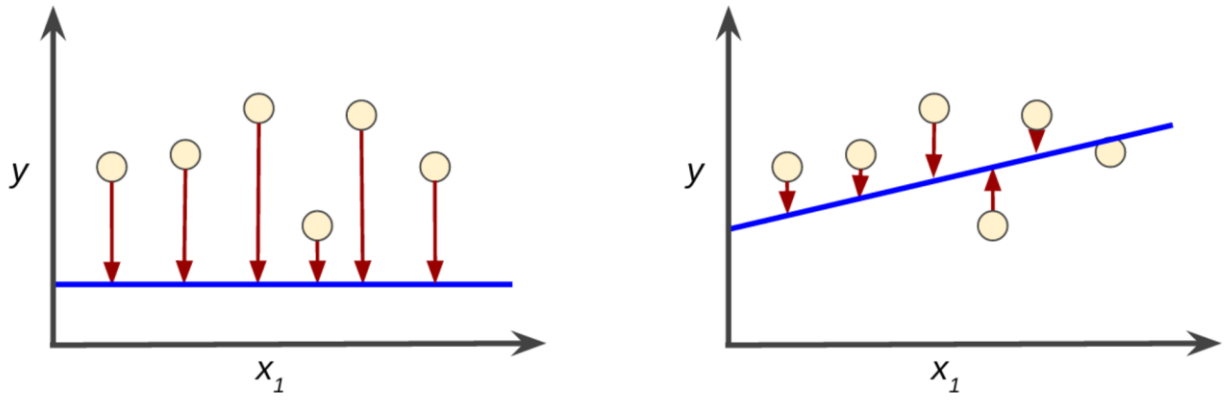
- Πόσο μεγάλο είναι το σύνολο δεδομένων: Όσο υψηλότερος είναι ο αριθμός των εικόνων στο σύνολο δεδομένων σας, τόσο περισσότερο θα χρειαστεί το μοντέλο να φτάσει σε ικανοποιητικά επίπεδα απόδοσης ανίχνευσης.
- Η πολυπλοκότητα των αντικειμένων που προσπαθείτε να εντοπίσετε: Προφανώς, εάν ο στόχος σας είναι να παρακολουθείτε μια μαύρη μπάλα πάνω σε λευκό φόντο, το μοντέλο θα συγκλίνει σε ικανοποιητικά επίπεδα ανίχνευσης αρκετά γρήγορα. Αν από την άλλη πλευρά, για παράδειγμα, θέλετε να εντοπίσετε πλοία σε λιμάνια, χρησιμοποιώντας κάμερες τότε η εκπαίδευση θα είναι μια πολύ πιο απαιτητική και χρονοβόρα διαδικασία, λόγω της υψηλής μεταβλητότητας του σχήματος και του μεγέθους πλοία, σε συνδυασμό με ένα πολύ δυναμικό υπόβαθρο.

4.7 Παρακολούθηση εκπαίδευσης και δείκτης Total Loss

Η εκπαίδευση ενός μοντέλου σημαίνει απλώς μάθηση (καθορισμός) καλών τιμών για όλα τα βάρη από τα επισημασμένα παραδείγματα (lampeled).

Στην εποπτευόμενη μάθηση, ένας αλγόριθμος μηχανικής μάθησης δημιουργεί ένα μοντέλο εξετάζοντας πολλά παραδείγματα και προσπαθώντας να βρει ένα μοντέλο που ελαχιστοποιεί την απώλεια. Αυτή η διαδικασία ονομάζεται εμπειρική ελαχιστοποίηση κινδύνου. Η απώλεια είναι η ποινή για μια κακή πρόβλεψη. Δηλαδή, η απώλεια είναι ένας αριθμός που δείχνει πόσο κακή ήταν η πρόβλεψη του μοντέλου σε ένα μόνο παράδειγμα. Εάν η πρόβλεψη του μοντέλου είναι τέλεια, η απώλεια είναι μηδέν. Διαφορετικά, η απώλεια είναι μεγαλύτερη.

Ο στόχος της εκπαίδευσης ενός μοντέλου είναι να βρει ένα σύνολο βαρών που έχουν χαμηλή απώλεια, κατά μέσο όρο, σε όλα τα παραδείγματα. Για παράδειγμα, το σχήμα δείχνει ένα μοντέλο υψηλής απώλειας στα αριστερά και ένα μοντέλο χαμηλής απώλειας στα δεξιά. Σημειώστε τα παρακάτω σχετικά με το σχήμα: Τα βέλη αντιπροσωπεύουν απώλεια. Οι μπλε γραμμές αντιπροσωπεύουν προβλέψεις.



Η παράμετρος total loss συνολική απώλεια δείχνει ότι όσο χαμηλότερη είναι η απώλεια, τόσο καλύτερο είναι το μοντέλο . Η απώλεια (total loss) υπολογίζεται κατά την εκπαίδευση και την επικύρωση και η διακοπή τους εκπαίδευσης αποφασίζεται όταν το μοντέλο ανταποκρίνεται καλά και για τους δυο αυτές παραμέτρους. Σε αντίθεση με την ακρίβεια, η απώλεια δεν είναι ποσοστό. Είναι ένα άθροισμα των σφαλμάτων που έγιναν για κάθε παράδειγμα σετ εκπαίδευσης

Στην περίπτωση των νευρωνικών δικτύων, η απώλεια είναι συνήθως αρνητική πιθανότητα log και υπολειπόμενο άθροισμα τετραγώνων για ταξινόμηση . Στη συνέχεια, φυσικά, ο κύριος στόχος σε ένα μοντέλο μάθησης είναι η μείωση (ελαχιστοποίηση) τους τιμής τους λειτουργίας απώλειας σε σχέση με τους παραμέτρους του μοντέλου, αλλάζοντας τους τιμές του διανύσματος βάρους μέσω διαφορετικών μεθόδων βελτιστοποίησης, τους η αναδρομή σε νευρωνικά δίκτυα.

Η τιμή απώλειας υποδηλώνει πόσο καλά ή κακά συμπεριφέρεται ένα συγκεκριμένο μοντέλο μετά από κάθε επανάληψη τους βελτιστοποίησης. Στην ιδανική περίπτωση, θα περίμενε κανείς τη μείωση τους απώλειας μετά από κάθε ή περισσότερες επαναλήψεις.

Η συνάρτηση απώλειας είναι αυτή που προσπαθεί να ελαχιστοποιήσει το SGD (Το πιο ευρέως γνωστό εργαλείο βελτιστοποίησης ονομάζεται Stochastic Gradient Descent,, ή πιο απλά, SGD.)[41] ενημερώνοντας επαναληπτικά τα βάρη στο νευρωνικό δίκτυο. Κατά τη διάρκεια της εκπαιδευτικής διαδικασίας, η απώλεια θα υπολογιστεί χρησιμοποιώντας τις προβλέψεις εξόδου του δικτύου και τις πραγματικές ετικέτες για την αντίστοιχη είσοδο.

Ας υποθέσουμε ότι το μοντέλο μας ταξινομεί εικόνες γατών και σκύλων και υποθέστε ότι η ετικέτα για τη γάτα είναι 0 και η ετικέτα για σκύλο είναι 1.

γάτα: 0

σκύλος: 1

[41] https://en.wikipedia.org/wiki/Stochastic_gradient_descent

Ας υποθέσουμε ότι περνάμε μια εικόνα μιας γάτας στο μοντέλο και η παρεχόμενη έξοδος είναι 0,25. Σε αυτήν την περίπτωση, η διαφορά μεταξύ της πρόβλεψης του μοντέλου και της πραγματικής ετικέτας είναι 0,25 - 0,00 = 0,25. Αυτή η διαφορά ονομάζεται το σφάλμα.

Αυτή η διαδικασία εκτελείται για κάθε έξοδο. Για κάθε εποχή, το σφάλμα συσσωρεύεται σε όλες τις μεμονωμένες εξόδους. Ας δούμε μια λειτουργία απώλειας που χρησιμοποιείται συνήθως στην πράξη που ονομάζεται μέσο τετράγωνο σφάλμα (MSE).

Mean Squared Error (MSE)

Για ένα μεμονωμένο δείγμα, με MSE, υπολογίζουμε πρώτα τη διαφορά (το σφάλμα) μεταξύ της παρεχόμενης πρόβλεψης εξόδου και της ετικέτας. Στη συνέχεια τετραγωνίζουμε αυτό το σφάλμα. Για μία μόνο είσοδο.

MSE (είσοδος) = (έξοδος - ετικέτα) (έξοδος - ετικέτα)

Εάν περάσαμε πολλά δείγματα στο μοντέλο ταυτόχρονα (μια παρτίδα δειγμάτων), τότε θα πάρουμε τον μέσο όρο των τετραγώνων σφαλμάτων σε όλα αυτά τα δείγματα. Αυτό απλώς απεικονίζει τα μαθηματικά πίσω από το πώς λειτουργεί μια λειτουργία απώλειας, το MSE. Υπάρχουν πολλές διαφορετικές λειτουργίες απώλειας με τις οποίες θα μπορούσαμε να εργαστούμε.

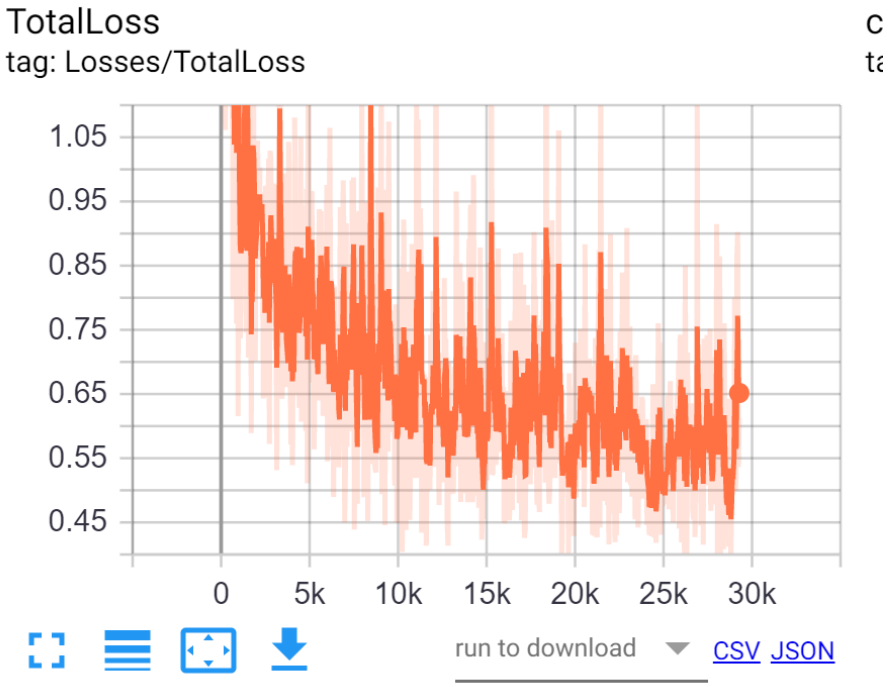
Η γενική ιδέα που μόλις δείξαμε για τον υπολογισμό του σφάλματος μεμονωμένων δειγμάτων θα ισχύει για όλους τους διαφορετικούς τύπους λειτουργιών απώλειας. Η εφαρμογή του τι κάνουμε πραγματικά με κάθε ένα από τα σφάλματα θα εξαρτάται από τον αλγόριθμο της δεδομένης συνάρτησης απώλειας που χρησιμοποιούμε.

Για παράδειγμα, υπολογίσαμε κατά μέσο όρο τα τετράγωνα σφάλματα για τον υπολογισμό του MSE, αλλά άλλες συναρτήσεις απώλειας θα χρησιμοποιήσουν άλλους αλγόριθμους για να προσδιορίσουν την αξία της απώλειας.

Στο δικό μας μοντέλο περάσαμε ολόκληρο το σετ προπόνησης ταυτόχρονα (batch_size = 1), τότε η διαδικασία που μόλις προχωρήσαμε για τον υπολογισμό της απώλειας θα συμβεί στο τέλος κάθε σετ κατά τη διάρκεια της εκπαίδευσης .

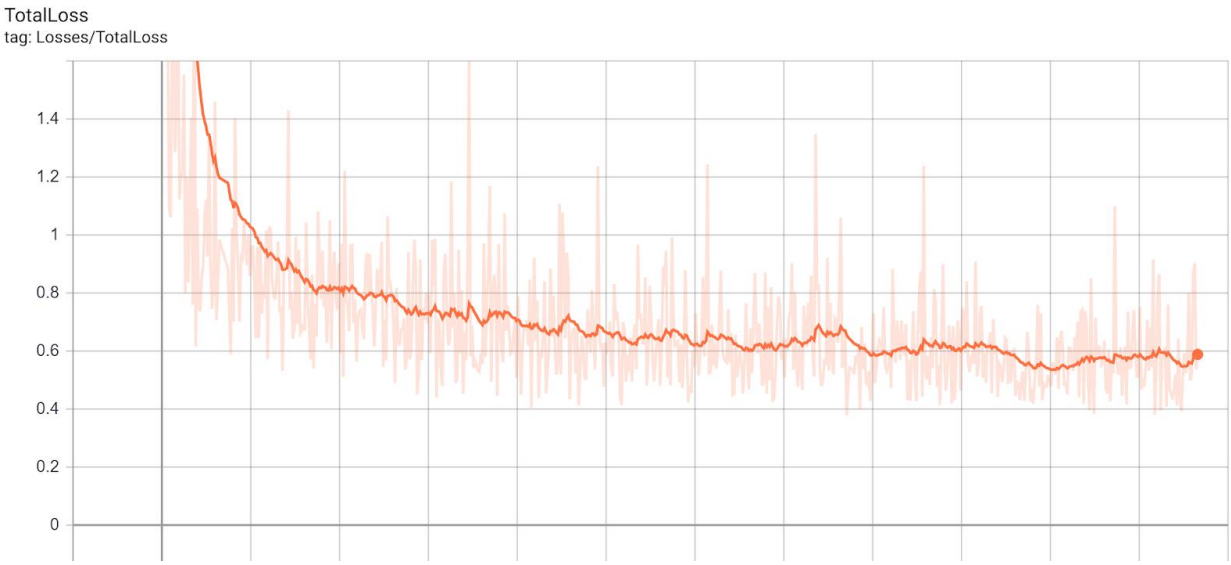
Εάν χωρίσουμε το σύνολο εκπαίδευσης μας σε παρτίδες και περάσαμε παρτίδες μία κάθε φορά στο μοντέλο μας, τότε η απώλεια θα υπολογιζόταν σε κάθε παρτίδα. Με οποιαδήποτε από τις δύο μεθόδους, καθώς η απώλεια εξαρτάται από τα βάρη, αναμένουμε να δούμε την τιμή της απώλειας να αλλάζει κάθε φορά που τα βάρη ενημερώνονται. Δεδομένου ότι ο στόχος της SGD είναι η ελαχιστοποίηση της απώλειας, θέλουμε να δούμε τη μείωση της απώλειας καθώς τρέχουμε περισσότερες παρτίδες.

Στην δική μας περίπτωση το μοντέλο επανεκπαιδεύτηκε με 500 εικόνες 1 παρτίδα εξολοκλήρου κινητών διαφόρων διαστάσεων σε διάστημα 23 ωρών. Από την εκπαίδευση του πειράματος προέκυψε το παρακάτω Total Loss του διαγράμματος .



Εικόνα 4-2:Total Loss-1

Κάνοντας ένα καλύτερο zoom φαίνονται οι τιμές καλύτερα , το γράφημα απώλειας του TensorBoard δείχνει ότι η απώλεια μειώθηκε σταθερά τόσο για την εκπαίδευση όσο και για την επικύρωση. Αυξάνοντας τα δεδομένα εισόδου σε διαφορετικές ομάδες διαστάσεων και δίνοντας μεγαλύτερο χρόνο εκπαίδευσης και με επανεκπαιδεύσεις τότε μπορεί να γίνει μια σύγκριση και η απώλεια να μειωνόταν στο 0,3 ή στο 0,2 για καλύτερη επίδοση του μοντέλου μας. Η τιμή που επιλέχθηκε η παύση της εκπαίδευσης ήταν στο 0,4 μετα από 23ωρες εκπαίδευσης.



Εικόνα 4-3:Total Loss 2

Ωστόσο για να δοκιμαστεί το μοντέλο απαιτείται να γίνει έλεγχος ότι κάνει αυτό για το οποίο εκπαιδεύτηκε, δηλαδή ότι ανιχνεύει τα αντικείμενα για τα οποία εκπαιδεύτηκε.

4.8 Ανίχνευση Αντικειμένου

Τα αποτελέσματα φαίνονται μέσα από αρχείο "Object_detection_tutorial.ipynb" Το αρχείο ανοίγει μέσα από το jupyter notebook. Το jupyter είναι μια δια δραστική διαδικτυακή εφαρμογή ανοικτού κώδικα που μας επιτρέπει την γραφή και εκτέλεση κώδικα σε πάνω από 40 γλώσσες, εδώ θα εκτελείτε σε python.

Εντοπίζετε το αρχείο Object_detection_tutorial.ipynb [42]. Από το μενού του jupyter Run-All εκτελείται ο κώδικας για την αναγνώριση αντικειμένων.

Τέλος, παρατίθενται τα αποτελέσματα της εγκατάστασης και καθώς η επεξεργασία γίνεται με remote session σε απομακρυσμένο υπολογιστή ο kernel δείχνει απασχολημένος για 2-3 λεπτά και έπειτα επιτρέπει την εμφάνιση των αποτελεσμάτων , οι εικόνες που λαμβάνουμε στο jupyter notebook είναι οι παρακάτω από το προ εκπαιδευμένο μοντέλο:

[42]https://pythonprogramming.net/introduction-use-tensorflow-object-detection-api-tutorial/

Εικόνα 4-4:Ανιχνευση Αντικειμένων 1

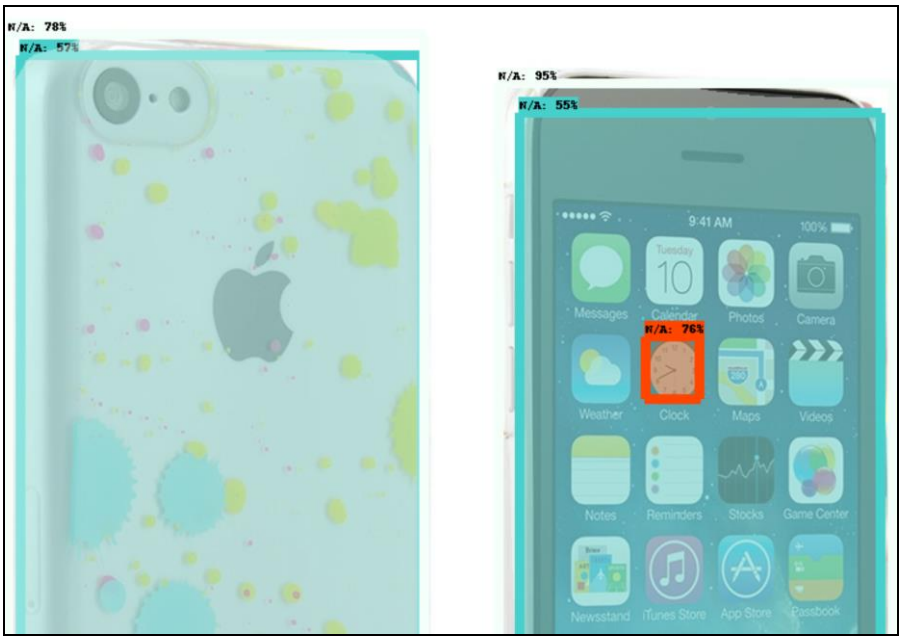
Από τον web browser πλέον μπορείτε να δείτε:

http://Host:8888/notebooks/object_detection_tutorial.ipynb

Ομοίως για το δικό μας μοντέλο η αναγνώριση των κινητών. Παρατηρείτε ότι η αναγνώριση φτάνει στο 85%.

Εικόνα 4-5: Ανίχνευση Αντικειμένων 2

Η δεύτερη εικόνα όπου η αναγνώριση φτάνει στο 95%. Στην εικόνα που είναι το πίσω μέρος του κινητού είναι λιγότερη η ευκρίνεια αλλά πάλι πραγματοποιείται αναγνώριση της τάξης 78%. Παρατηρούμε ότι ακόμη και τα μέσα πλαίσια των κινητών αναγνωρίζονται και αυτά.



Εικόνα 4-6: Ανίχνευση Αντικειμένων 3

Κάποια από τα πρώτα αποτελέσματα σε frequency CPU: 205110000Hz είναι 0.20511Ghz.

και η κατανάλωση μνήμης: 669081600, περίπου 638.09B που δεν υπερβαίνει 10% της ελεύθερης μνήμης του συστήματος.

Στα επόμενα κεφάλαια θα προσπαθήσουμε να μετρήσουμε και να μελετήσουμε για το δικό μας πείραμα με τα κινητά, τόσο την ταχύτητα επεξεργασίας, την μνήμη και την κατανάλωση ενέργειας. Στα μελλοντικά συμπεράσματα θα δούμε με ποια εργαλεία μπορεί να γίνει optimization του μοντέλου για καλύτερα αποτελέσματα.

4.9 Αποθηκευμένο μοντέλο

Όσον αφορά τα δεδομένα του μοντέλου, αποθηκεύονται με τη μορφή βαρών, τα οποία χρησιμοποιούνται ως αρχικά βάρη στο προ-εκπαιδευμένο μοντέλο, και κατά συνέπεια αναπροσαρμόζονται μετά την προσθήκη του δικού μας συνόλου δεδομένων και την ολοκλήρωση της επανεκπαίδευσης. Δεν απαιτεί την εκτέλεση του αρχικού κώδικα δημιουργίας μοντέλου, γεγονός που το καθιστά χρήσιμο για κοινή χρήση ή ανάπτυξη.

Η πρόοδος του μοντέλου μπορεί να αποθηκευτεί κατά τη διάρκεια - και μετά την εκπαίδευση. Αυτό σημαίνει ότι ένα μοντέλο μπορεί να συνεχίσει από εκεί που σταμάτησε και να αποφύγει τους μεγάλους χρόνους προπόνησης. Η αποθήκευση σημαίνει επίσης ότι μπορείτε να μοιραστείτε το μοντέλο σας και άλλοι μπορούν να αναδημιουργήσουν την εργασία σας. Κατά τη δημοσίευση ερευνητικών μοντέλων και τεχνικών, οι περισσότεροι επαγγελματίες μηχανικής μάθησης μοιράζονται: κώδικα για τη δημιουργία του μοντέλου και τα εκπαιδευμένα βάρη ή παραμέτρους για το μοντέλο. Η κοινή χρήση αυτών των δεδομένων βοηθά τους άλλους να κατανοήσουν πώς λειτουργεί το μοντέλο και να τα δοκιμάσουν μόνοι τους με νέα δεδομένα.

Το SavedModel CLI βοηθά να εξεταστεί το μοντέλο και να ελεγχθεί το SignatureDef του μοντέλου. Δίνει λεπτομέρειες σχετικά με την είσοδο Tensor dtype και το σχήμα στο οποίο εκπαιδεύτηκε το μοντέλο. Έτσι, πρέπει να δοθεί η κατάλληλη είσοδος στο μοντέλο με την ίδια μορφή και να ληφθεί η έξοδος όπως ορίζεται από το SignatureDef's.

5 Εργαστηριακές Μετρήσεις Πειράματος

5.1 Πειραματικό Περιβάλλον

Η αποτελεσματικότητα του μοντέλου φαίνεται όταν αυτό γίνεται ανάπτυξη και μπαίνει σε παραγωγική διαδικασία, συνεπώς για να μελετηθεί και να καταγραφούν οι μεταβλητές απόκρισης του θα πρέπει ανάπτυξη του μοντέλου το οποίο θα πρέπει να μπορέσει να δεχθεί αιτήματα requests για την αναγνώριση των αντικειμένων. Αυτή η διαδικασία επιλέχθηκε να γίνει με τα εργαλεία TensorFlow-Serving[43] και Docker.

Το μοντέλο έχει εκπαιδευτεί στο σύνολο δεδομένων των εικόνων "κινητά τηλέφωνα" και λαμβάνει μια εικόνα JPEG ως είσοδο και επιστρέφει την αναγνώριση της εικόνας.

Η πειραματική μας αξιολόγηση στοχεύει στην εκτίμηση της σκοπιμότητας της αναγνώρισης αντικειμένων. Για το σκοπό αυτό πραγματοποιήθηκαν στο εργαστηριακό περιβάλλον μετρήσεις διαφορετικών συνόλων.

Οι μετρήσεις δοκιμών για την ανταπόκριση του μοντέλου απαιτούν έναν client ο οποίος θα στέλνει τις αντίστοιχες εικόνες προς αναγνώριση στο μοντέλο. Για τον λόγο αυτό δημιουργήθηκε ένα python script το οποίο εξυπηρετεί αυτόν τον σκοπό. Επίσης στο ίδιο script καταγράφονται και οι CPU και Memory Usage τις συγκεκριμένης διεργασίας για κάθε αριθμό αιτημάτων ξεχωριστά.

Επίσης απαιτείται να γίνει serving το μοντέλο ώστε να μπορεί να δεχτεί τα αιτήματα ως υπηρεσία.

5.1.1 Εξυπηρέτηση Μοντέλων

Η εξυπηρέτηση (serving) μοντέλων μηχανικής εκμάθησης γρήγορα και εύκολα είναι μια από τις βασικές προκλήσεις κατά τη μετάβαση από την πειραματική διαδικασία στην παραγωγή. Η εξυπηρέτηση μοντέλων μηχανικής εκμάθησης είναι η διαδικασία λήψης ενός εκπαιδευμένου μοντέλου και η διάθεσή του για την εξυπηρέτηση αιτημάτων πρόβλεψης. Για το σκοπό αυτό, ένας από τους ευκολότερους τρόπους εξυπηρέτησης μοντέλων μηχανικής εκμάθησης είναι με τη χρήση του TensorFlow Serving with Docker. Το Docker είναι ένα εργαλείο που συσκευάζει λογισμικό σε μονάδες που ονομάζονται κοντέινερ και περιλαμβάνει όλα όσα χρειάζονται για την εκτέλεση του λογισμικού.

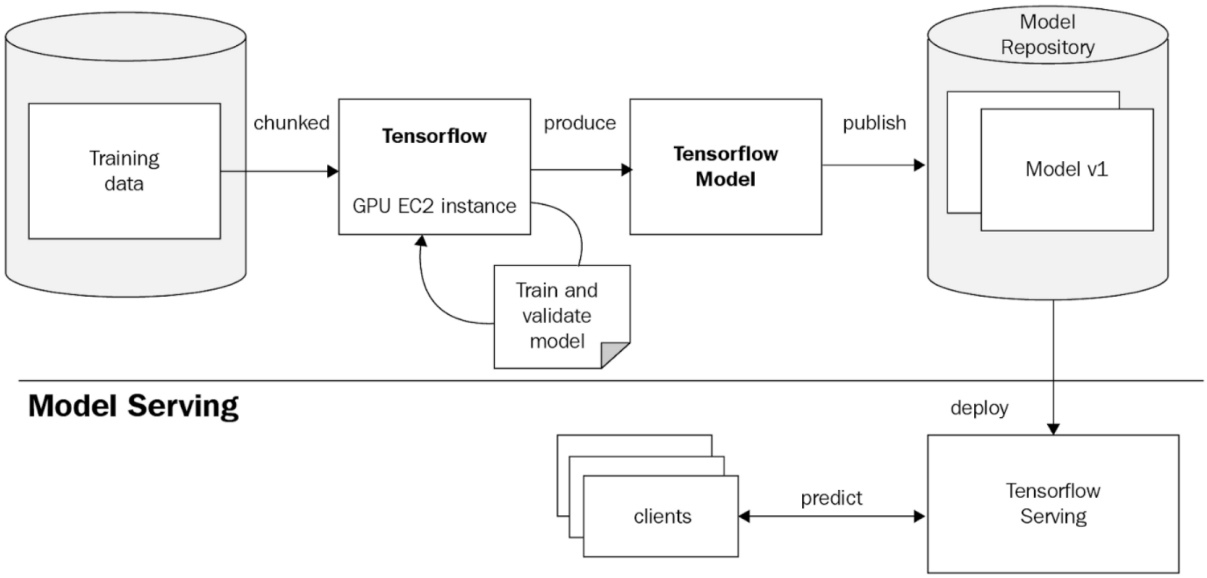
[43] https://www.tensorflow.org/tfx/guide/serving

Όπως και σε όλα τα στάδια του πειράματος έτσι και εδώ θα χρησιμοποιηθεί το TensorFlow-Serving[44], ένα σύστημα που εξυπηρετεί μοντέλα μηχανικής μάθησης που είναι επίσης διαθέσιμο στο cloud και μέσω ανοιχτού κώδικα. Είναι εξαιρετικά ευέλικτο όσον αφορά τους τύπους πλατφορμών ML που υποστηρίζει και τους τρόπους για ενοποίηση με συστήματα που μεταφέρουν νέα μοντέλα από την εκπαίδευση στο serving. Ταυτόχρονα, ο βασικός πυρήνας του κώδικα αναζητά και εξάγει συμπεράσματα βελτιστοποιημένα για το μοντέλο για να αποφευχθούν παγίδες απόδοσης που παρατηρήθηκαν στις υλοποιήσεις.

Στο παραπάνω διάγραμμα φαίνεται μια σύντομη εικόνα για το πώς μοιάζει όλη αυτή η διαδικασία, από την κατασκευή του μοντέλου έως την εξυπηρέτηση αυτού του μοντέλου σε ένα τελικό σημείο χρησιμοποιώντας το Tensorflow Serving. Η καλύτερη επιλογή για την εξυπηρέτηση των περισσότερων τύπων μοντέλων μπορεί να είναι η εκτέλεση ενός κεντρικού μοντέλου σε έναν διακομιστή όπου μπορεί να ζητήσει από οποιοδήποτε είδος συσκευής, μπορεί να ζητήσει από επιτραπέζιους υπολογιστές, κινητές συσκευές ή ενσωματωμένες συσκευές. Τότε ο διακομιστής θα έκανε την εξαγωγή για εσάς και θα επέστρεφε τις προβλέψεις. Από αυτήν την πρόβλεψη, μπορείτε να την αποδώσετε σε οποιαδήποτε συσκευή.

Ένα μεγάλο πλεονέκτημα σε αυτόν τον τύπο αρχιτεκτονικής είναι ότι ας υποθέσουμε ότι έχετε αρκετούς πελάτες που έχουν πρόσβαση στο τελικό σημείο που είναι συγκεντρωμένο σε έναν διακομιστή. Όλοι έχουν πρόσβαση στην ίδια έκδοση μοντέλου και μπορούν να απολαύσουν την ενημερωμένη έκδοση χωρίς προβλήματα, ενώ η κλιμάκωση για αιτήματα μεγάλου αριθμού γίνεται εύκολη προσθέτοντας απλώς εξισορροπητές φορτίου. Θα ήταν δύσκολο αν είχατε αναπτύξει το μοντέλο σε κάθε υπολογιστή-πελάτη, επομένως η διαχείριση εκδόσεων και η τοποθέτηση νέων ενημερώσεων γίνεται δύσκολη.

[44] https://arxiv.org/abs/1712.06139



Εικόνα 5-1: Model Serving[45]

Τα παραπάνω διάγραμμά δείχνει μια σύντομη εικόνα για το πώς μοιάζει όλη αυτή η διαδικασία, από την κατασκευή του μοντέλου έως την εξυπηρέτηση αυτού του μοντέλου σε ένα τελικό σημείο χρησιμοποιώντας το Tensorflow Serving. Η καλύτερη επιλογή για την εξυπηρέτηση των περισσότερων τύπων μοντέλων μπορεί να είναι η εκτέλεση ενός κεντρικού μοντέλου σε έναν διακομιστή όπου μπορεί να ζητήσει από οποιοδήποτε είδος συσκευής, μπορεί να ζητήσει από επιτραπέζιο, κινητό ή ένα φορητό υπολογιστή. Τότε ο διακομιστής θα έκανε την επεξεργασία για εσάς και θα επέστρεφε τις προβλέψεις. Αυτή η πρόβλεψη, μπορεί να αποδοθεί σε οποιαδήποτε συσκευή. Ένα μεγάλο πλεονέκτημα σε αυτόν τον τύπο αρχιτεκτονικής είναι ότι ας υποθέσουμε ότι έχετε αρκετούς πελάτες που έχουν πρόσβαση στο τελικό σημείο που είναι συγκεντρωμένο σε έναν διακομιστή. Όλοι έχουν πρόσβαση στην ίδια έκδοση μοντέλου, ενώ η κλιμάκωση για μεγάλα αιτήματα γίνεται εύκολη προσθέτοντας απλώς εξισορροπητές φορτίου. Θα ήταν δύσκολο αν είχατε αναπτύξει το μοντέλο σε κάθε υπολογιστή-πελάτη, επομένως η διαχείριση εκδόσεων και η τοποθέτηση νέων ενημερώσεων θα γινόταν δύσκολα.

Εκπαιδεύτηκε ένα μοντέλο νευρωνικού δικτύου για την αναγνώριση εικόνων κινητών (βασισμένο στο COCO), εξοικονομώντας εκπαιδευμένο μοντέλο με βάση το TensorFlow και θα εξυπηρετηθεί με το Tensorflow Serving και με το Docker[46].

```
# Start TensorFlow Serving container and open the REST API port
docker run -t --rm -p 8501:8501 \
    -v "$TESTDATA/ cellphone_inference_graph:/models/cellphone" \
    -e MODEL_NAME= cellphone \
```

[45] Image is taken from Packt

[46] https://www.docker.com/

```
tensorflow/serving &
```

5.2 Περιγραφή πειράματος

Το service εξυπηρετείτε σε απομακρυσμένο υπολογιστή με διεύθυνση http://host:8501/v1/models/cellphone:predict , ο edge computer είναι ένα Linux ubuntu 4.4.0 GNU/Linux:

VCPUs: 4

RAM: 8 GB

Πιο αναλυτικά τα χαρακτηριστικά του στον πίνακα.

CPU op-mode(s):	32-bit, 64-bit	BogoMIPS:	4197.36
Byte Order:	Little Endian	Hypervisor vendor:	Xen
CPU(s):	6	Virtualization type:	full
Vendor ID:	GenuineIntel		
CPU family:	6		
Model name:	Intel(R) Xeon(R) Silver 4110 CPU @ 2.10GHz		
CPU MHz:	2095.110		

Στην παρούσα πειραματική μελέτη γίνεται μια παραδοχή η οποία αφορά τα χαρακτηριστικά του Η/Υ. Χρησιμοποιήθηκε ένα virtual machine του οποίου τα χαρακτηριστικά θα μπορούσαν να είναι ένας edge computer. Επιπρόσθετα για τις ανάγκες των αιτημάτων χρησιμοποιήθηκε φορητή συσκευή με προδιαγραφές ισχύος που αντίστοιχές υπάρχουν σε φορητές συσκευές τελευταίας τεχνολογίας όπως smartphones.

Η πειραματική μας αξιολόγηση στοχεύει στην αξιολόγηση της σκοπιμότητας της εκφόρτωσης αναγνώρισης αντικειμένων σε έναν εικονικό server. Για το σκοπό αυτό, χρησιμοποιούμε τη μέτρηση της συνολικής απόκρισης χρόνου.

Από τη φορητή συσκευή σε περιβάλλον ubuntu τρέχει το python script[47] το οποίο καλεί την υπηρεσία που εκτελείται η αναγνώριση αντικειμένων η οποία φιλοξενείται στην υποδομή στον server. Αυτό το σενάριο θα κατεβάσει μια εικόνα και θα την στείλει στον διακομιστή επανειλημμένα ενώ μετρά τους χρόνους απόκρισης

Το script αποστέλλει ταυτόχρονα διαφορετικές εικόνες ως σύνολα αιτημάτων και επιστρέφει τον μέσο όρο για τον χρόνο απόκρισης των αιτημάτων για την αναγνώριση, την

[47] Appendix

κατανάλωση επεξεργαστικής ισχύ και την κατανάλωση μνήμης. Στόχος της αποστολής των συγκεκριμένων αιτημάτων είναι η καταγραφή του χρόνου απόκρισης του μοντέλου για την αναγνώριση των συγκεκριμένων εικόνων. Επιπρόσθετα, για τα ίδια αιτήματα καταγράφετε η επεξεργαστική ισχύος και μνήμη που απαιτείται για να πραγματοποιηθεί η αναγνώριση των συγκεκριμένων εικόνων. Η αποτύπωση της υποδομής και του πειράματος παρουσιάζεται στο διάγραμμα της εικόνας Εικόνα 5 1: Model Serving

Για τις ανάγκες του πειράματος τα αιτήματα χωρίστηκαν σε σύνολα τεσσάρων κατηγοριών ταυτόχρονων αιτημάτων, αυξανόμενα με βήμα 1 έως 50, αιτήματα αυξανόμενα με βήμα 5 έως 100, αιτήματα αυξανόμενα με βήμα 10 έως 200 και τέλος τυχαίος αριθμός 50 ταυτόχρονων αιτημάτων χωρίς συγκεκριμένο βήμα. Ο παραπάνω διαχωρισμός πραγματοποιήθηκε για να γίνει διάκριση και αποτύπωση της συμπεριφοράς του μοντέλου και της υποδομής του edge cloud στους δείκτες χρόνου απόκρισης, CPU και RAM διαφορετικών συνόλων μικρών, μεσαίων και μεγαλύτερων αιτημάτων.

Για το συγκεκριμένο πείραμα εστάλησαν 7.500 αιτήματα από την φορητή συσκευή τα οποία χωρίστηκαν σε ομάδες αποστολών κι καταγράφηκαν τα παρακάτω αποτελέσματα όπως φαίνονται στα διαγράμματα.

5.3 Δείκτες Απόδοσης

Ρύθμιση απόδοσης: Στόχοι και παράμετροι

Για την βελτιστοποίηση του TensorFlow Serving, υπάρχουν συνήθως 2 τύποι στόχων 3 ομάδες παραμέτρων για να βελτιώσετε αυτούς τους στόχους.

Στόχοι

Το Serving είναι ένα διαδικτυακό σύστημα εξυπηρέτησης για εκπαιδευόμενα μοντέλα μηχανικής μάθησης. Όπως με πολλά άλλα διαδικτυακά συστήματα εξυπηρέτησης, πρωταρχικός στόχος απόδοσης είναι η μεγιστοποίηση της απόδοσης, διατηρώντας παράλληλα την καθυστέρηση κάτω από ορισμένα όρια. Ανάλογα με τις λεπτομέρειες και την ωριμότητα των αιτημάτων, μπορεί να δοθεί έμφαση περισσότερο για το μέσο λανθάνοντα χρόνο από ό, τι η καθυστέρηση, αλλά κάποια έννοια του λανθάνοντος χρόνου και της απόδοσης είναι συνήθως οι μετρήσεις βάσει των οποίων ορίζονται οι στόχοι απόδοσης. Ο χρόνος απόκρισης και η απόδοση του edge computer.

Έτσι 3 ομάδες παραμέτρων διαμορφώνου την απόδοση 1) το μοντέλο 2) τα αιτήματα 3) ο διακομιστής φιλοξενίας (υλικό & λογισμικό).

5.3.1 Χρόνος Απόκρισης (Response Time)

Η απόδοση του μοντέλου στον κεντρικό υπολογιστή (CPU), στη συσκευή (GPU) ή σε συνδυασμό τόσο του κεντρικού υπολογιστή όσο και της συσκευής, μπορεί να παρακολουθηθεί με διάφορα εργαλεία. Το προφίλ του μοντέλου βοηθά να γίνει κατανοητή η κατανάλωση πόρων υλικού (χρόνος και μνήμη) των διαφόρων λειτουργιών TensorFlow (ops) του μοντέλου και να επιλύσετε τα σημεία συμφόρησης επιδόσεων και, τελικά, να κάνετε το μοντέλο να εκτελείται πιο γρήγορα.

Το μοντέλο είναι αυτό που καθορίζει τον υπολογισμό χρόνου που θα εκτελέσει η υπηρεσία TensorFlow Serving κατά τη λήψη κάθε εισερχόμενης αίτησης. Αυτό σημαίνει ότι τα μοντέλα καθορίζουν τον χρόνο επεξεργασίας της εικόνας σε ms καθώς την ποιότητα αναγνώρισης της εικόνας. Το TensorFlow Serving χρησιμοποιεί τον χρόνο εκτέλεσης TensorFlow για να κάνει την αναγνώριση των αιτημάτων. Αυτό σημαίνει ότι ο μέσος χρόνος καθυστέρησης της εκτέλεσης ενός αιτήματος με την υπηρεσία TensorFlow Serving είναι συνήθως τουλάχιστον αυτός του υπολογίζεται απευθείας με το TensorFlow.

Αυτό σημαίνει ότι εάν σε ένα δεδομένο μηχάνημα, η εξαγωγή σε ένα μόνο παράδειγμα διαρκεί 2 δευτερόλεπτα και ο στόχος για την καθυστέρηση είναι στο ένα δευτερόλεπτο, πρέπει να σχεδιαστούν αιτήματα και πως αυτά ανταποκρίνονται στο TensorFlow και τα δευτερεύοντα γραφήματα του μοντέλου σε αυτόν τον λανθάνοντα χρόνο και επανασχεδιαστεί το μοντέλο με λανθάνουσα καθυστέρηση ως σχεδιαστικό περιορισμό. Πρέπει να ληφθεί υπόψη ότι, ενώ ο μέσος λανθάνων χρόνος εκτέλεσης συμπερασμάτων με την υπηρεσία TensorFlow Serving συνήθως δεν είναι χαμηλότερος από τη χρήση του TensorFlow απευθείας, όπου το TensorFlow Serving διατηρεί την ουρά καθυστέρησης για πολλά αιτήματα, ενώ ταυτόχρονα χρησιμοποιεί αποτελεσματικά το υποκείμενο υλικό για τη μεγιστοποίηση της απόδοσης .

Για την απόδοση του μοντέλου και της συσκευής υπάρχουν διάφοροί χρόνοι σε ms που λαμβάνουν χώρα οι οποίοι μπορούν να υπολογιστούν με διαφορετικά εργαλεία αυτοί για πιο ξεκάθαρη εικόνα γίνονται αναγωγή για ένα server φιλοξενίας του μοντέλου και για ένα πυρήνα: Είναι ο χρόνος εισόδους της εικόνας, ο χρόνος εξόδου, ο χρόνος υπολογισμού που κάνει η υπηρεσία να ανταποκριθεί, άλλοι χρόνοι, συμπεριλαμβανομένου του Python, γενικά ο χρόνος απόκρισης του server για την επεξεργασία, συνολικά χαρακτηρίζεται σαν χρόνο απόκρισης και για τα πειράματα μας λήφθηκε ο μέσος όρος της απόκρισης αυτών των αιτημάτων.

5.3.2 Μνήμη (Memory)

Η παρακολούθηση της κατανάλωσης μνήμης μπορεί να γίνει με το πως αυτή κατανέμεται σε συγκεκριμένο χρονικό διάστημα κατά την αποστολή συγκεκριμένων αιτημάτων.

- Allocation Κατανομή - Ο αριθμός των κατανομών μνήμης που πραγματοποιήθηκαν κατά τη διάρκεια του διαστήματος προφίλ
- Deallocation - Ο αριθμός των απενεργοποιήσεων μνήμης στο διάστημα προφίλ
- Χωρητικότητα μνήμης (Memory Capacity) - Η συνολική χωρητικότητα
- Μέγιστη χρήση μνήμης (Peak Heap Usage)- Η μέγιστη χρήση μνήμης στο διάστημα που γίνεται η παρακολούθηση. Αυτό το πεδίο περιέχει και άλλα χαρακτηριστικά για μεγαλύτερη ανάλυση όπως
    - Χρονική σήμανση (Timestamp) - Η χρονική σήμανση του πότε σημειώθηκε η μέγιστη χρήση της μνήμης στο γράφημα λωρίδας χρόνου
    - Stack Reservation - Ποσότητα μνήμης που διατηρείται στη στοίβα
    - Κατανομή σωρού (Stack Reservation) - Ποσότητα μνήμης που διατίθεται στο σωρό
    - Ελεύθερη μνήμη (Free Memory) - Ποσότητα ελεύθερης μνήμης
    - Η χωρητικότητα μνήμης είναι το συνολικό άθροισμα της κράτησης στοίβας, της κατανομής σωρών και της ελεύθερης μνήμης.
    - Κατακερματισμός (Fragmentation)- Το ποσοστό κατακερματισμού (το χαμηλότερο είναι καλύτερο). Υπολογίζεται ως ποσοστό (1 - Μέγεθος του μεγαλύτερου τμήματος της ελεύθερης μνήμης / Συνολική ελεύθερη μνήμη)

Η χρήσης της μνήμης στην συσκευή μπορεί επίσης να καταγραφεί και να αναλυθεί ωστόσο και σ αυτή την περίπτωση μετρήσαμε το μέσο όρο μνήμης κατανάλωσης μνήμης που απαιτήθηκε για την επεξεργασία των αιτημάτων.

5.3.3 Επεξεργαστική ισχύος (CPU Usage)

Οι δοκιμές απόδοσης του TensorFlow βασίστηκαν στον χρόνο και τη μνήμη που απαιτείται για την εκπαίδευση του μοντέλου σε έναν τύπο μονάδας επεξεργασίας αυτόν της CPU. Επιπλέον, υπάρχουν τεχνικοί περιορισμοί της μελέτης όλο το πείραμα εκτελέστηκε σε περιβάλλον Linux όπως αναφέρθηκε πιο πάνω.

Η μηχανική μάθηση, και τα νευρωνικά δίκτυα ειδικότερα, συχνά πρέπει να επεξεργαστούν μια μεγάλη ποσότητα δεδομένων εκπαίδευσης για ακριβείς υπολογισμούς. Ιδιαίτερα οι υπολογισμοί απαιτούν μεγάλη ποσότητα επεξεργασίας. Με τη σημερινές

αρχιτεκτονικές μονάδες επεξεργασίας δεδομένων, έχουμε την ικανότητα να εκπαιδεύσουμε ακόμη μεγαλύτερα μοντέλα. Οι μονάδες επεξεργασίας αποτελούνται από τρία κύρια συστατικά: μνήμη, αριθμητική λογική μονάδα και μονάδα ελέγχου. Στη μνήμη αναφερθήκαμε στην προηγουμένη παράγραφο, ενώ η αριθμητική λογική μονάδα, ή ALU, επεξεργάζεται αριθμητικές και λογικές πράξεις.

Τέλος, η μονάδα ελέγχου είναι η ροή των οδηγιών μεταξύ της ALU, της κύριας μνήμης και των συσκευών I / O. Ο συνδυασμός αυτών των συστατικών καθορίζει τα χαρακτηριστικά των μονάδων επεξεργασίας. Τα δύο κύριες μονάδες επεξεργασίας σύγχρονων υπολογιστών είναι η CPU και η GPU. Κάθε υπολογιστής έχει ένα κύριο συστατικό για την εκτέλεση αριθμητικής λογικής και τον έλεγχο, την κεντρική μονάδα επεξεργασίας (CPU).

Η κύρια λειτουργία της CPU πρέπει να εκτελεί διαδοχικά οδηγίες που διατηρούνται στη μνήμη του υπολογιστή . Η CPU έχει ουσιαστικό ρόλο στον υπολογισμό των νευρωνικών δικτύων δεδομένου ότι επεξεργάζεται τους γενικούς αριθμητικούς υπολογισμούς κατά τη φάση της προπόνησης. Οι CPU συνήθως κατασκευάζονται με μερικούς ισχυρούς πυρήνες επεξεργασίας ιδανική για εκτέλεση διαδοχικών εργασιών. Επιπλέον, όλα τα I/O στη φάση της εκπαίδευσης, όπως η φόρτωση δεδομένων εκπαίδευσης, αντιμετωπίζονται με τη χρήση CPU ανεξάρτητα από τη χρήση της GPU για υπολογισμό.

Οι επιδόσεις του TensorFlow, τόσο σε CPU όσο και σε GPU, δοκιμάστηκαν με μέτρηση του χρόνου και της κατανομής μνήμης κατά τη φάση εκπαίδευσης των νευρωνικών δικτύων καθώς τα γραφήματα ροής δεδομένων TensorFlows χρησιμοποιούνται κυρίως για τη βελτιστοποίηση του χρόνου εκπαίδευσης. Η εκπαίδευση είναι μια επαναληπτική διαδικασία όπου το μοντέλο δοκιμάζεται και στη συνέχεια άλλαξε ώστε να μπορεί να δοκιμαστεί ξανά. Προσδιορίστηκαν οι εκτελέσεις της εκπαίδευσης του μοντέλου οι οποίες έγιναν σε CPU είτε σε GPU, στο δικό μας πείραμα λήφθηκαν αποτελέσματα για την CPU.

Κατά τις επαναλήψεις αποστολής των αιτημάτων στο δικό μας μοντέλο ελήφθησαν και οι μετρήσεις κατανάλωσης CPU και οι μετρήσεις αποτυπώθηκαν σε διάγραμμα.

Θα μπορούσε να αποτελέσει ξεχωριστή μελέτη η απόδοση ενός μοντέλου σε CPU και GPU καθώς η αποστολή των αιτημάτων γίνεται στους ίδιους χρόνους με τις ίδιες επαναλήψεις .

5.3.4 Κατανάλωση Ενέργειας (Energy Consumption)

Η κατανάλωση ενέργειας έχει μελετηθεί ευρέως στον τομέα της αρχιτεκτονικής των υπολογιστών. Υπάρχουν πολλά εργαλεία παρακολούθησης ισχύος και απόδοσης. Μία επισκόπηση των εργαλείων που μπορούν να διευκολύνουν την μέτρηση της ισχύος χαρακτηρίζονται με τα ακόλουθα κριτήρια:

1. Γλωσσική διεπαφή του εργαλείου: Πώς μπορεί να προσπελαστεί και να χρησιμοποιηθεί το εργαλείο. Έχει παράσχει ο προγραμματιστής του εργαλείου ένα σύνολο επιλογών διασύνδεσης γραμμής εντολών (CLI) ή ενός γραφικού περιβάλλοντος εργασίας χρήστη (GUI).

2. Υποστηριζόμενα λειτουργικά συστήματα: Υπάρχει υποστήριξη για διαφορετικούς τύπους λειτουργικών συστημάτων. MacOS (M), Linux (L) και Windows (W).

3. Φιλικό προς το χρήστη: Αυτό εξαρτάται από τη διεπαφή γλώσσας που παρέχεται και τη διαθεσιμότητα τεκμηρίωσης του εργαλείου.

4. Ωριμότητα: Η διάρκεια από την καθιέρωση του εργαλείου και η τεχνική υποστήριξη για το εργαλείο.

5. Έρευνα ή εμπορικό προϊόν: Είναι το εργαλείο αποτέλεσμα ερευνητικής προσπάθειας ή εμπορικού προϊόντος.

Τα περισσότερα εργαλεία που υπάρχουν στην αγορά Powertop, Powerstat, and Power Statistics programs αναφέρονται στις μετρήσεις σε φορητές συσκευές. Όπως επίσης και το gnome-power-statistics είναι ένα πρόγραμμα gui για την υποδομή διαχείρισης ισχύος. Επιτρέπει στους χρήστες να απεικονίζουν την κατανάλωση ενέργειας του υλικού φορητού υπολογιστή. Για server ή desktop υπολογιστές προτείνετε να τοποθετηθεί ένας ηλεκτρονικός μετρητής watt ο οποίος προσαρμόζεται στη συσκευή. Πιο συγκεκριμένα στη παρούσα περίπτωση που το πείραμα χρησιμοποιεί επεξεργαστή Intel(R) Xeon(R) Silver 4110 CPU @2.10GHz υπάρχουν εξειδικευμένα εργαλεία σύμφωνα με την intel [48]τα οποία πάλι είναι ανάλογα με την αρχιτεκτονική του υλικού (hardware).

Όσον αφορά το software που υπάρχει διαθέσιμο για τον υπολογισμό της κατανάλωσης ενέργειας αυτό δεν μπορεί να γίνει με αξιόπιστο τρόπο ειδικά σε servers καθώς δεν μπορεί να υπολογιστεί ανά διεργασία , ή ανά επεξεργαστή. Κάποιες εντολές[49] μπορούν να μας πουν πόση ενέργεια αντλείτε από την μπαταρία και αφορούν μόνο φορητούς υπολογιστές. Αλλά από την έρευνα που έγινε δεν εντοπίστηκε κατάλληλο πρόγραμμα για desktop υπολογιστές.

Επιπρόσθετα ερευνήθηκε η δυνατότητα δημιουργίας κώδικα υπολογισμού κατανάλωσης ενέργειας με τη βοήθεια βιβλιοθήκης energyusage 0.0.14[50] ωστόσο υπάρχει και εδώ περιορισμός. Λόγω των μεθόδων με τις οποίες γίνεται η μέτρηση ενέργειας (μέσω της διεπαφής Intel RAPL και του NVIDIA-smi), το πακέτο είναι διαθέσιμο μόνο σε πυρήνες Linux που έχουν τη διεπαφή RAPL ή / και μηχανήματα με GPU Nvidia.

[48] https://software.intel.com/content/www/us/en/develop/blogs/measuring-application-power-consumption-on-linux-operating-system.html

[49] https://askubuntu.com/questions/13337/how-do-i-measure-server-power-consumption

[50] https://pypi.org/project/energyusage/

Τέλος, έχουν ερευνηθεί και api toolkit τα οποία απευθύνονται κυρίως σε φορητές συσκευές ή συσκευές με rapl interface[51].

[51] http://web.eece.maine.edu/~vweaver/projects/rapl/

5.4 Αποτελέσματα Πειραμάτων

Η πειραματική μας αξιολόγηση στοχεύει στην εκτίμηση της σκοπιμότητας της αναγνώρισης αντικειμένων και καταλήγει σε μια υποδομή ενός Η/Υ που θα μπορούσε να αποτελέσει μία συσκευή με χαμηλές προδιαγραφές σε σχέση με αυτές που προσφέρει μία υποδομή σε cloud όπως έγινε στην πειραματική μελέτη του COSMOS.

Για το σκοπό αυτό, χρησιμοποιήθηκε η μέτρηση του συνολικού χρόνου απόκρισης, οι κατανάλωση πόρων σε επεξεργαστική (CPU) και σε μνήμη (memory). Για τον πειραματισμό, δημιουργήθηκε ένα σύνολο δεδομένων που αποτελείται από πολλά αιτήματα. Πάνω από 7000 αιτήματα ανίχνευσης αντικειμένων, τα οποία ενεργοποιούνται σε τακτά χρονικά διαστήματα. Τελικά καταγράφηκε η απόδοση του υπολογιστή που βρίσκεται η υπηρεσία αναγνώρισης, στέλνοντας σύνολα δεδομένων χαμηλών αιτημάτων της τάξης του **Α1 και Α2 σύνολα (1-50),** υψηλών αυξανόμενων αιτημάτων **Β σύνολο (50-200)** καθώς και σύνολα τα οποία δεν είχαν κάποια αυξανόμενη σειρά αιτημάτων αλλά **τυχαία σύνολο Γ**.

Σε όλες τις περιπτώσεις παρατηρήθηκε ο χαμηλός χρόνος απόκρισης σε τιμές από 0.072s στον μικρότερο αριθμό αιτημάτων και την μεγαλύτερη στα τυχαία δείγματα 2.938 και μέσο όρο χρόνου απόκρισης 0.721s διαγράμματα(1-4) .

Έτσι, σε μια σχεδόν γραμμική αύξηση το οι βασικές μετρήσεις θεωρούνται ικανοποιητικές, μας δίνει αυξανόμενο χρόνο απόκρισης, ενώ στην τυχαία αποστολή αιτημάτων παρατηρείτε μια διακύμανση του χρόνου με μεγάλες αποκλείσεις, αυτό να οφείλεται καθώς δεν υπάρχει η αρχιτεκτονική του COSMOS με τα VNF Instances όπως επίσης ο ελεγκτής COSMOS αποτελείται από τα ακόλουθα κύρια στοιχεία: πρόβλεψη φόρτου εργασίας, εξισορρόπηση φορτίου, έλεγχος εισαγωγής και παρακολούθηση. Η πρόβλεψη φόρτου εργασίας χρησιμοποιεί το Kalman Filter όπως είδαμε στο κεφάλαιο 3 για να εκτιμήσει τον αριθμό των αιτήσεων και να αυξήσει τη λειτουργία του ελέγχου εισαγωγής, το οποίο απορρίπτει αιτήματα που δεν μπορούν να αντιμετωπιστούν από τις παρουσίες εντοπισμού αντικειμένων (που εφαρμόζονται χρησιμοποιώντας το TensorFlow) στο cloud edge.

Θεωρώντας τον χρόνο απόκρισης ως μέτρηση που ουσιαστικά μεταφράζεται σε συνδυασμό υπολογισμών και χρόνων μετάδοσης, μπορούμε να συμπεράνουμε αυτόν τον υπολογισμό ο χρόνος είναι ο κυρίαρχος παράγοντας της συνολικής εκτέλεσης της εφαρμογής. Όσον αφορά τις διακυμάνσεις των τιμών, αποδίδεται στη διαφορετική πολυπλοκότητα των υπολογισμών, ανάλογα με κάθε μεμονωμένη εικόνα που αποστέλλεται στην εφαρμογή.

Σχετικά με το performance του μηχανήματος και την χρήση της CPU απαιτείται μια αυξημένη χρήση της CPU όπως παρατηρείται από τα γραφήματα τις τάξης 78.9% -98% αυτό γίνεται καθώς για τους σκοπούς της επεξεργασίας (computation) για την ανίχνευση του αντικειμένου. Η απόδοση του TensorFlow Serving εξαρτάται σε μεγάλο βαθμό από την

εφαρμογή που εκτελεί, το περιβάλλον στο οποίο αναπτύσσεται και το λογισμικό με το οποίο μοιράζεται την πρόσβαση στους υποκείμενους πόρους υλικού. Επίσης το μοντέλο καθορίζει τον υπολογισμό που θα εκτελέσει η υπηρεσία TensorFlow Serving κατά τη λήψη κάθε εισερχόμενης αίτησης, οπότε ένα μοντέλο αναγνώρισης αντικειμένων απαιτεί μεγάλη υπολογιστική ισχύ.

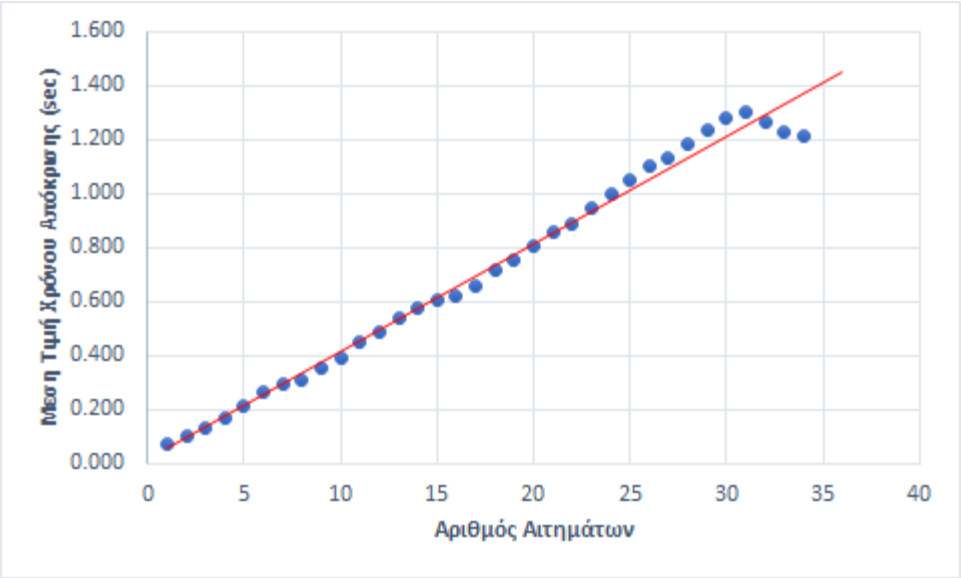
Παρατηρείται ότι το CPU utilization φτάνει στο όριο των πόρων (διαγράμματα 5-12)αυτό σημαίνει ότι πρέπει είτε να βελτιωθεί η αποδοτικότητα των εισερχόμενων αιτημάτων (για παράδειγμα, αποφεύγοντας τον περιττό υπολογισμό) είτε να μειωθεί ο υπολογισμός εκφόρτωσης, να δημιουργηθεί εξισορρόπηση φορτίου. Δεδομένα που είναι διαθέσιμα στους ελεγκτές του COSMOS όπως αναφέρθηκε και πιο πάνω.

Τέλος, αναφορικά με την μνήμη του μηχανήματος παρατηρούμε ότι παραμένει σε χαμηλά επίπεδα για όλα τα σετ δεδομένων με μέσο όρο διακύμανσης 3.32MB-4.01MB διαγράμματα (5-12) Στην εξυπηρέτηση ML, το μέγεθος του μοντέλου έχει σημασία, οπότε τα μικρότερα μοντέλα χρησιμοποιούν λιγότερη μνήμη, λιγότερο χώρο αποθήκευσης και εύρος ζώνης δικτύου και φορτώνουν γρηγορότερα. Σε ορισμένες περιπτώσεις, περιορισμοί στη μνήμη υλικού ή περιορισμοί υπηρεσιών ενδέχεται να επιβάλλουν όριο στο μέγεθος του μοντέλου.

5.4.1 Μετρήσεις Χρόνου Απόκρισης

Για τον πειραματισμό, δημιουργήσαμε ένα σύνολο δεδομένων που αποτελείται από μικρές ομάδες αιτημάτων ανίχνευσης αντικειμένων και επιστρέφει το μέσο χρόνο απόκρισης του Server.

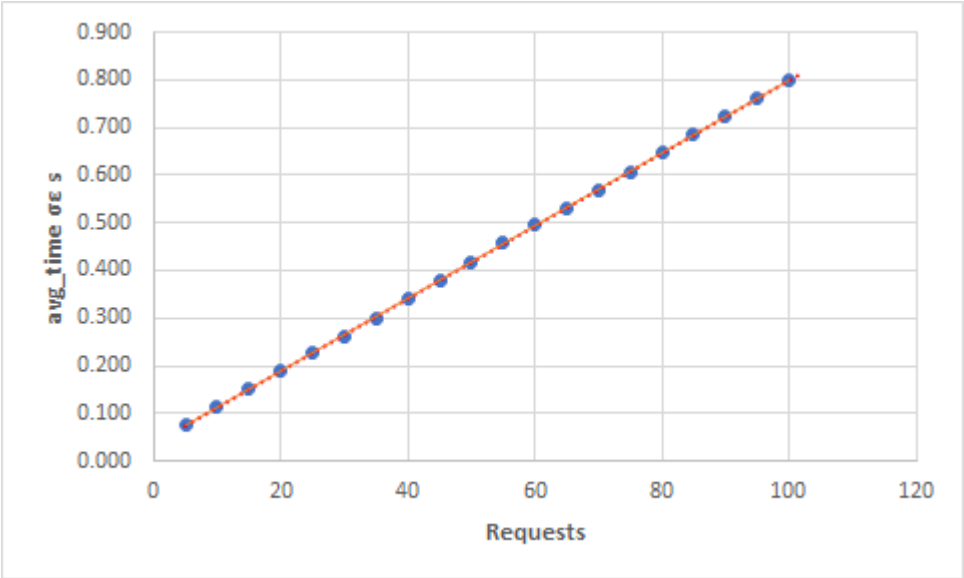
Η εκτέλεση των αιτημάτων γίνεται με αυξανόμενο ρυθμό από έναν πίνακα τυχαίων χαμηλών τιμών με βήμα 1:



Γράφημα 5-1: Χρόνος Απόκρισης - Σύνολο αιτημάτων Α1

ο χρόνος απόκρισης σε s, είναι ανάλογος με τα αιτήματα , ο ελάχιστος χρόνος που παρατηρείται είναι 0.072 στα λιγότερα αιτήματα και ο μεγαλύτερος 1.212 . Ωστόσο ο χρόνος απόκρισης δεν κυμαίνεται με μεγάλες διάφορες.

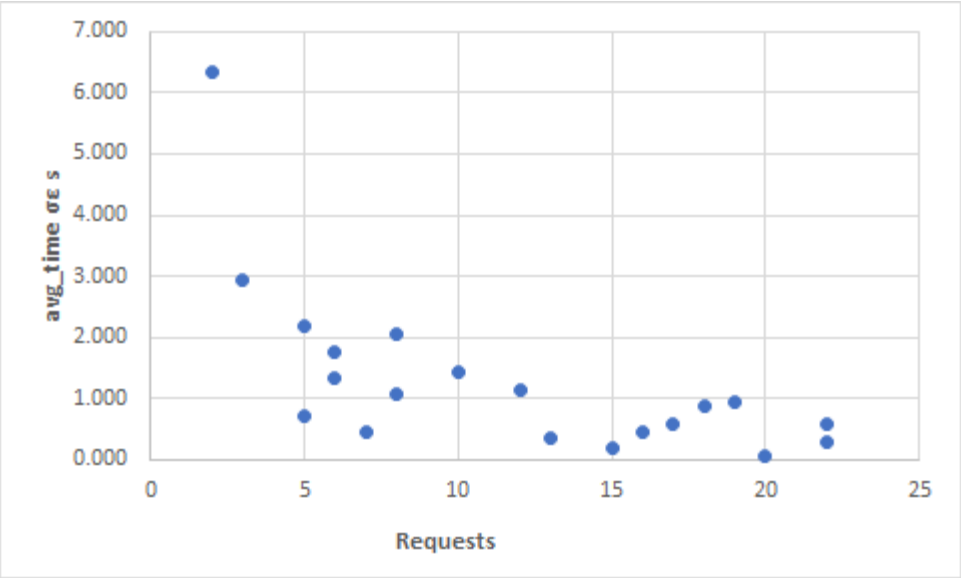
Ένα δεύτερο σύνολο σταθερό βήμα (5) αυξανόμενων αιτημάτων μας δίνει την ίδια εικόνα σταδιακής αύξησης του χρόνου απόκρισης με χρόνους απόκρισης 0.077s και ο μεγαλύτερος 0.797 s.



Γράφημα 5-2:Χρόνος Απόκρισης - Σύνολο αιτημάτων Α2

Ωστόσο ένα πιο πραγματικό πείραμα θα ήταν τα τυχαία αιτήματα στο server, όχι με έναν συνεχώς αυξανόμενο ρυθμό αιτημάτων καθώς σε μία πραγματική αποστολή εικόνων για ανίχνευση αντικειμένου από κινητά τηλέφωνα δε σημαίνει ότι θα είναι σταδιακά αυξανόμενα τα αιτήματα αλλά με τυχαία αποστολή.

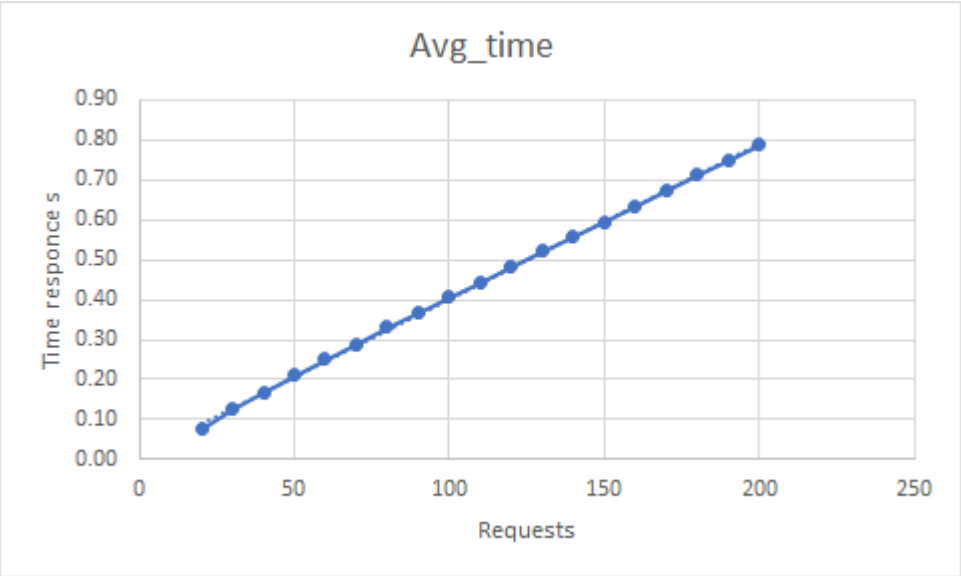
Οπότε σε τυχαίο σύνολο δεδομένων όπως φαίνεται στο επόμενο διάγραμμα



Γράφημα 5-3: Χρόνος Απόκρισης - Σύνολο αιτημάτων Β

Εδώ ο χρόνος απόκρισης παραμένει χαμηλός με τον μικρότερο χρόνο στο 0.075s στα 20 αιτήματα και τον μέσο όρο των χρόνων του συνόλου των δεδομένων να μη ξεπερνάει τα 1.287s.

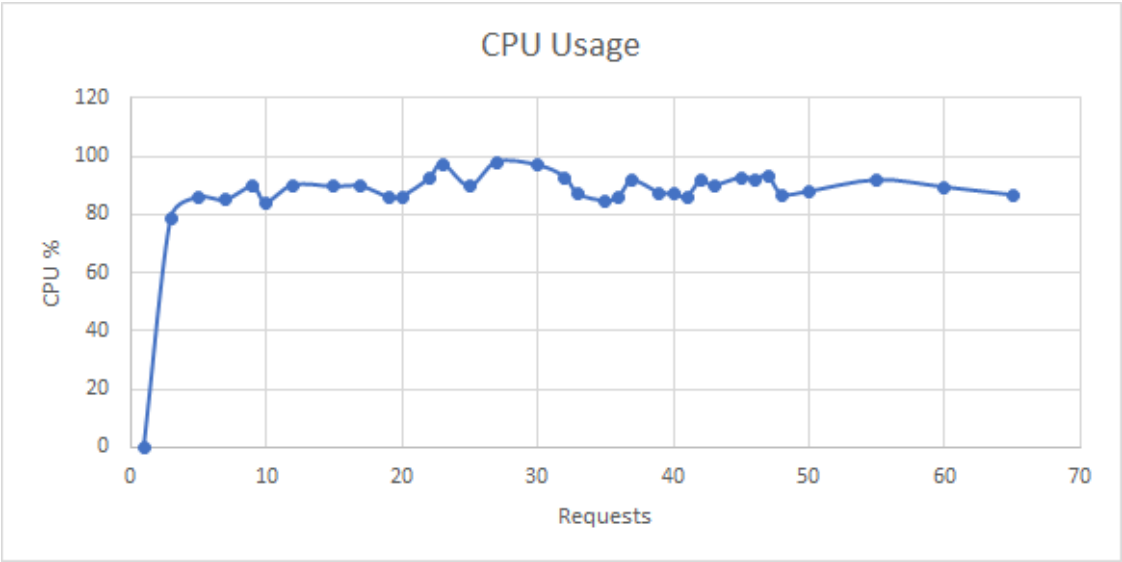
Τέλος, σε μία μέτρηση με υψηλό αριθμό αιτημάτων βαθμιαία αυξανόμενο παρατηρήθηκε ίδια συμπεριφορά του μοντέλου, αυξανόμενος ρυθμός απόκρισης των αιτημάτων ωστόσο ο χρόνος απόκρισης παραμένει μικρός πχ για τα 200 αιτήματα είναι μόλις 0.8s



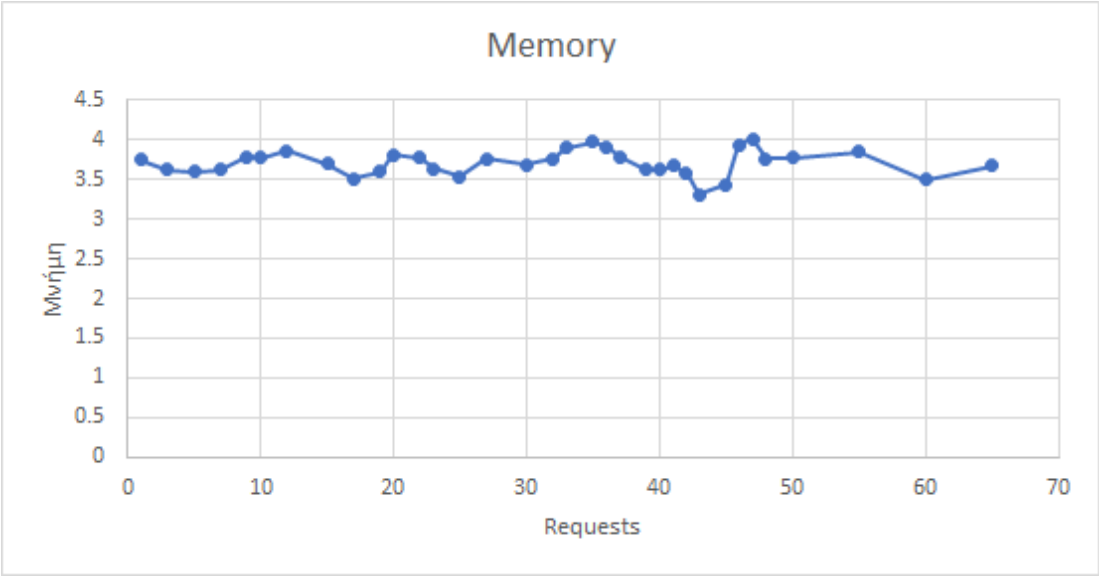
Γράφημα 5-4:Χρόνος Απόκρισης - Σύνολο αιτημάτων Γ

5.4.2 Μετρήσεις Επεξεργαστικής Ισχύος και Μνήμης

Στις ίδιες μετρήσεις και για το πρώτο σύνολο δεδομένων που πραγματοποιήθηκαν οι χρόνοι απόκρισης καταγράφηκε και το CPU και Memory usage για την διεργασία Tensorflow οπότε εξήχθησαν τα παρακάτω διαγράμματα και συμπεράσματα. Άρα για το σύνολο των δεδομένων που αντικατοπτρίζει ένα αυξανόμενο τρόπο αιτημάτων σε σχέση με την συνολική χρήση της CPU στο επι τοις 100 ποσοστό είναι

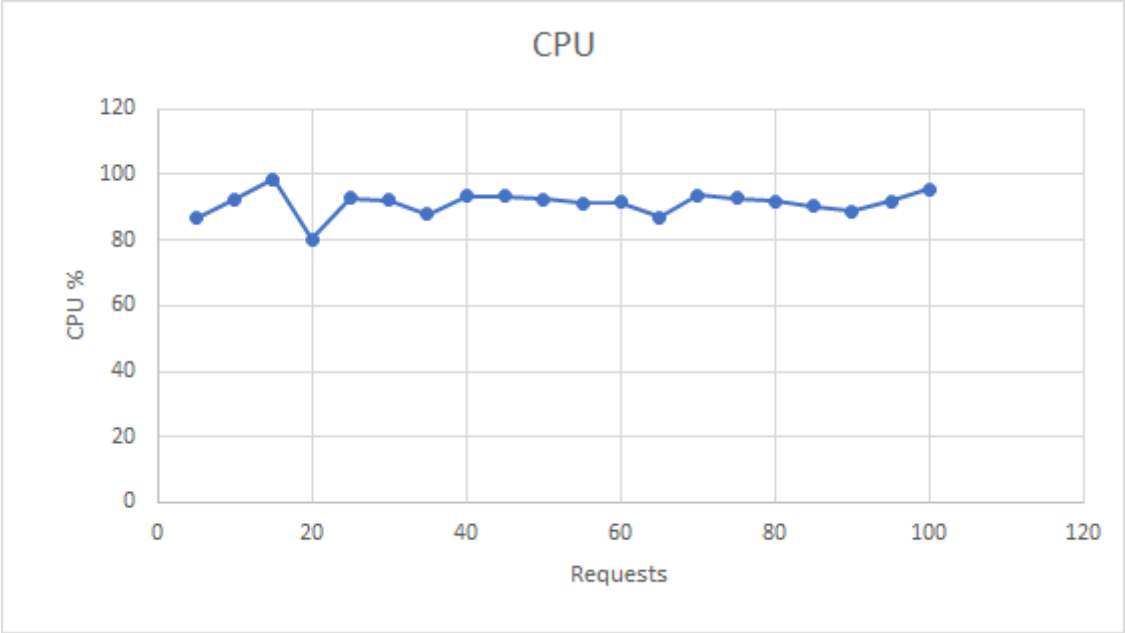


Γράφημα 5-5:CPU- Σύνολο αιτημάτων Α1

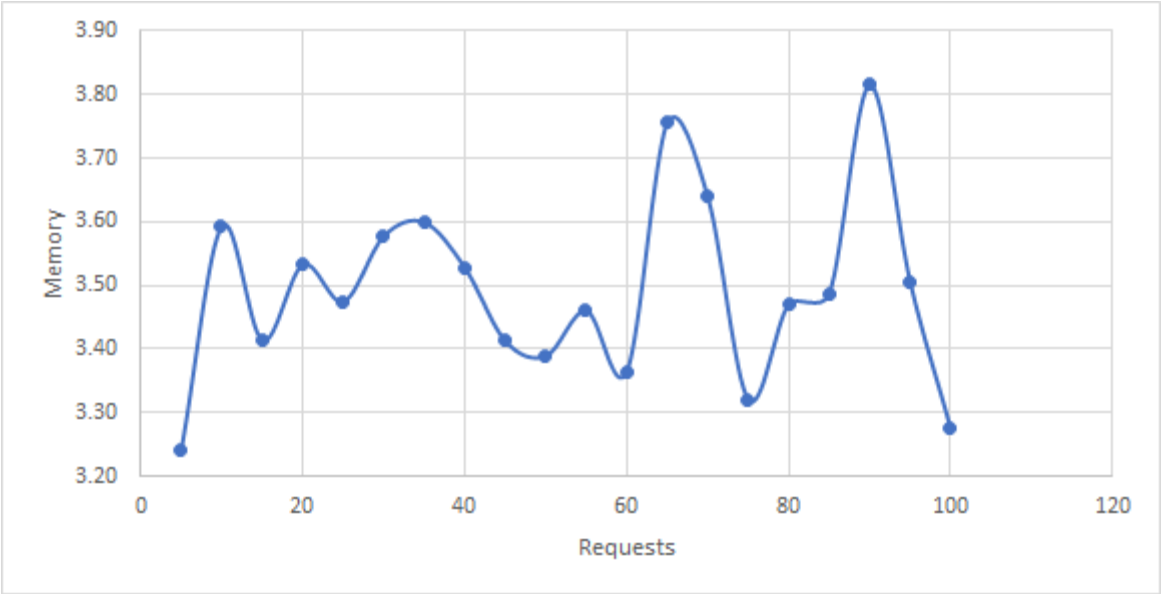


Γράφημα 5-6:Memory- Σύνολο αιτημάτων Α1

Εδώ παρατηρείτε μια αύξηση στην επεξεργασία οπότε η CPU κυμαίνεται σε επίπεδα από 78% έως 97%. Από την άλλη η μνήμη παραμένει σε χαμηλά επίπεδα από 3.31MB μέχρι 4.01 MB. Ομοίως συμβαίνει με CPU/μνήμη και με την σταθερά αυξανόμενη τιμή αιτημάτων ανά 5.

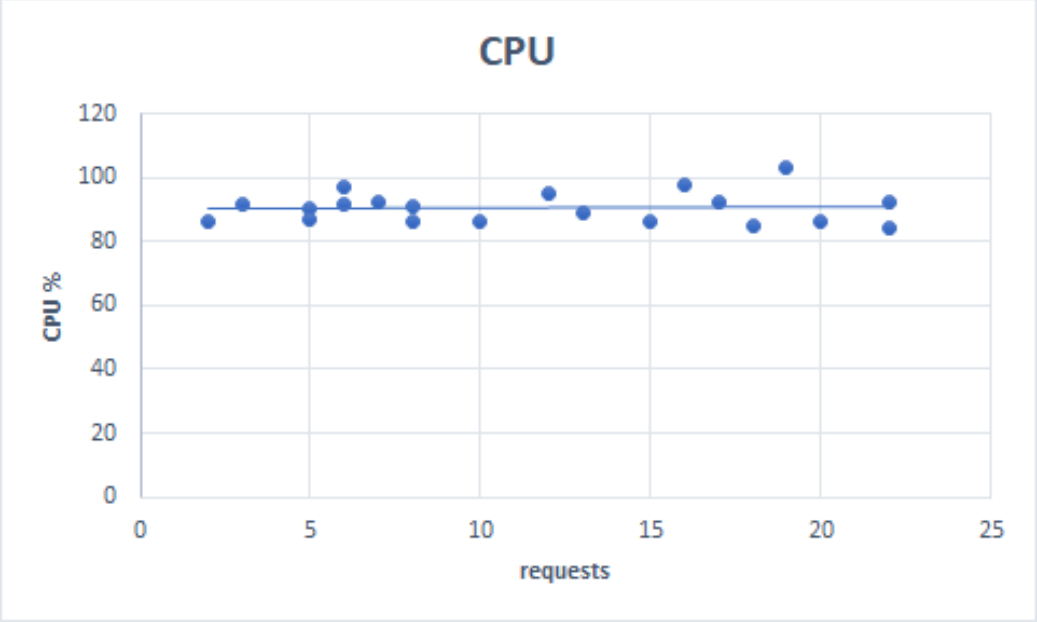


Γράφημα 5-7:CPU - Σύνολο αιτημάτων Α2

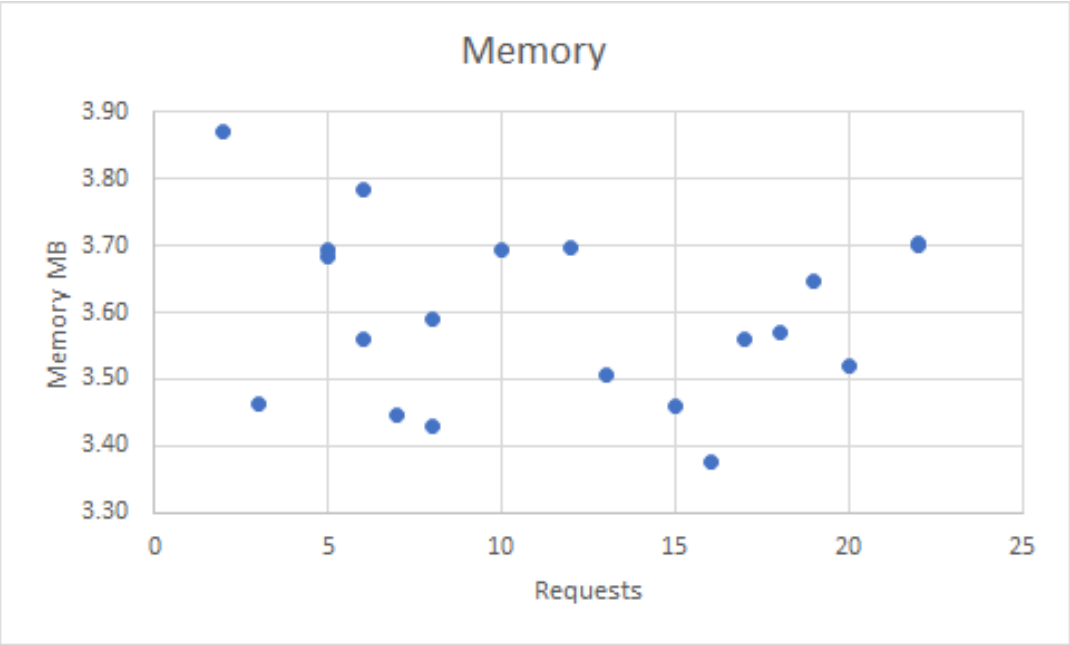


Γράφημα 5-8:Memory -σύνολο αιτημάτων A2

Την ίδια συμπεριφορά CPU και Memory έχουν και τα τυχαία δείγματα μη σταθερά αυξανόμενα αιτήματα που ελήφθησαν για συγκεκριμένα χρονικά διαστήματα. Η ίδια συμπεριφορά παρατηρείται και σε αυτά τα σετ δεδομένων από 84% -99% η CPU και 3.38MB έως 3.87MB η μνήμη.

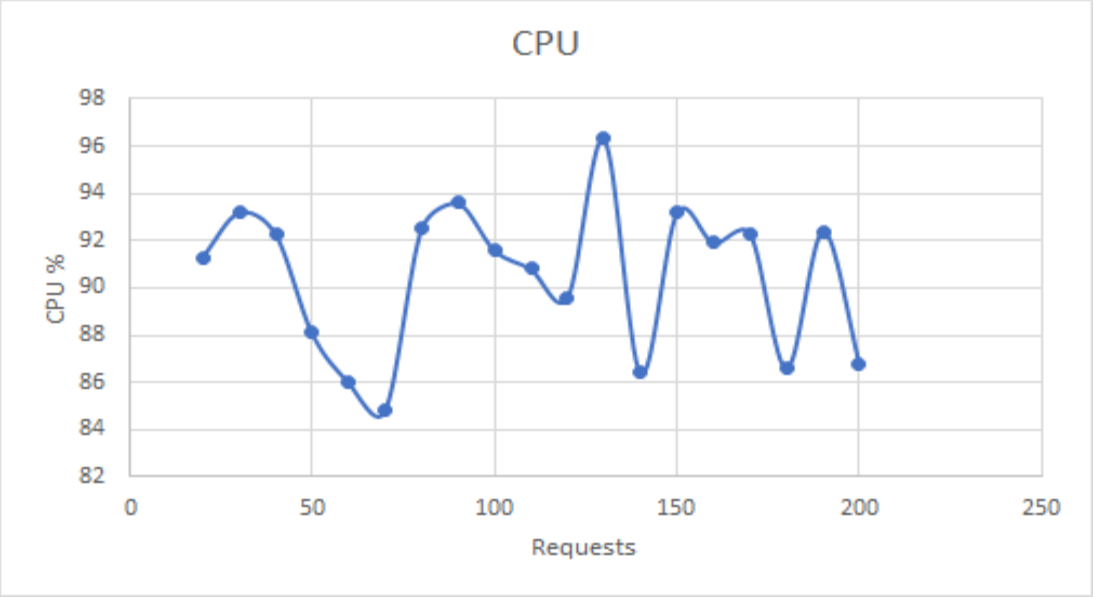


Γράφημα 5-9:CPU - Σύνολο Αιτημάτων Γ

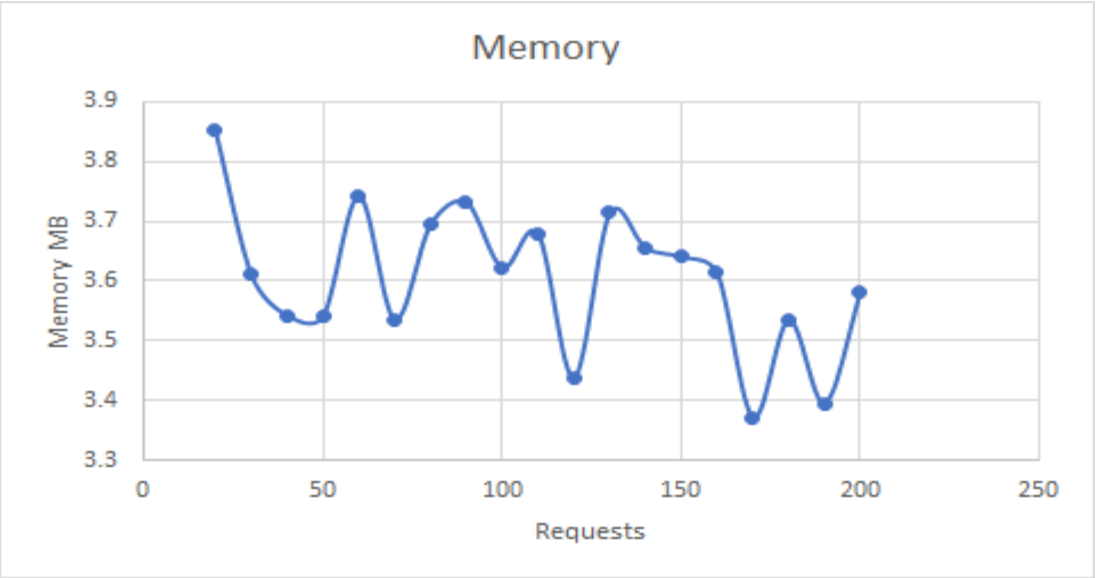


Γράφημα 5-10: Memory -Σύνολο αιτημάτων Γ

Τέλος, σε σετ δεδομένων με μεγάλο αριθμό requests υπάρχει η ίδια συμπεριφορά 84.8% -96.3% και αντίστοιχα η μνήμη 3.37MB -3.85MB.



Γράφημα 5-11: CPU -Σύνολο αιτημάτων Β



Γράφημα 5-12:Memory- Σύνολο αιτημάτων B

6 Συμπεράσματα-Μελλοντικές Ενέργειες

Όπως ήδη αναπτύχθηκε στα αρχικά κεφάλαια οι Τεχνολογίες Πληροφορικής και Επικοινωνιών (ΤΠΕ) διεισδύουν ολοένα και περισσότερο στη ζωή των ανθρώπων, συνυπάρχουν σχεδόν σε κάθε προϊόν και υπηρεσία και είναι ελάχιστες οι παραγωγικές εργασίες που πραγματοποιούνται πλέον χωρίς τη συμβολή τους. Το σύστημα αναγνώρισης αντικειμένων μπορεί να εφαρμοστεί πλέον σε πολλά πεδία, σε αναγνώριση προσώπου, ανίχνευση σφαλμάτων, αναγνώριση χαρακτήρων, αναγνώριση κίνησης, πρόσφατα παρουσιάστηκε βίντεο social distancing recognition[52] για να εξυπηρετήσει ανάγκες λόγω covid-19.

Η επεξεργασία και η αναγνώριση εικόνων έχουν εξελιχθεί με πολλές και ισχυρές εφαρμογές, όπως η ασφάλεια και η παρακολούθηση. Επιπρόσθετα ακόμη και στην ιατρική και έχει δημιουργηθεί μια μεγάλη αξία από επιχειρηματική άποψη. Λειτουργίες αναγνώρισης σχήματος, όπως αναγνώριση προσώπου ή αντικειμένων, οπτική γεωγραφική τοποθεσία, ανάγνωση γραμμωτού κώδικα και αυτοματοποιημένη βοήθεια οδηγού, μεταξύ άλλων λειτουργιών που σχετίζονται με τον αυτοματισμό της βιομηχανίας, έχουν δείξει την ευελιξία αυτής της τεχνολογίας.

Σε συνδυασμό με το AI (Artificiant Intelligent), αυτή η τεχνολογία έχει αρχίσει να δημιουργεί πολύτιμες ευκαιρίες ανάπτυξης σε πολλούς κλάδους, όπως παιχνίδια, κοινωνική δικτύωση και ηλεκτρονικό εμπόριο. Για παράδειγμα, το Twitter και το Facebook, δύο μεγάλες πλατφόρμες στον κόσμο της κοινωνικής δικτύωσης, έχουν επωφεληθεί από την τεχνολογία όσον αφορά την αφοσίωση του κοινού καθώς έχουν δημιουργήσει μια πιο συνδεδεμένη εμπειρία ενθαρρύνοντας τους χρήστες να μοιράζονται εικόνες και να επισημαίνουν στους φίλους τους.

Η έλευση των ψηφιακών φωτογραφικών μηχανών, ιδίως των φωτογραφικών μηχανών που είναι ενσωματωμένες σε smartphone, έχει οδηγήσει σε εκθετική αύξηση του όγκου του ψηφιακού περιεχομένου με τη μορφή εικόνων και βίντεο. Μια τεράστια ποσότητα οπτικών και ψηφιακών δεδομένων συλλαμβάνεται και κοινοποιείται μέσω πολλών εφαρμογών, ιστότοπων, κοινωνικών δικτύων και άλλων ψηφιακών καναλιών.

Αρκετές επιχειρήσεις έχουν αξιοποιήσει αυτό το διαδικτυακό περιεχόμενο για να παρέχουν καλύτερες και εξυπνότερες υπηρεσίες στους πελάτες τους, με τη χρήση ψηφιακής επεξεργασίας εικόνας. Για παράδειγμα, τον Οκτώβριο του 2019, η SnapPay Inc., ένας πάροχος πλατφόρμας πληρωμών με έδρα τις ΗΠΑ, ξεκίνησε την τεχνολογία πληρωμής αναγνώρισης προσώπου στην περιοχή της Βόρειας Αμερικής. Χρησιμοποιώντας αυτήν την τεχνολογία στη λύση πληρωμών της, η εταιρεία στοχεύει να επιτρέψει στους πελάτες της ένα νέο επίπεδο ευκολίας για πληρωμές σε καταστήματα λιανικής.

52 https://www.techprofree.com/social-distancing-detector-in-python-with-deep-learning-source-code/

Ωστόσο, ο καθαρισμός δεδομένων και η ισχύος επεξεργασίας υλικού παραμένουν ως οι δύο σημαντικές προκλήσεις που εμπλέκονται στην οικοδόμηση αξιόπιστης τεχνολογίας. Επίσης, λαμβάνοντας υπόψη τον χρόνο, την πολυπλοκότητα και το κόστος που σχετίζονται με την ανάπτυξη λογισμικού για την αναγνώριση εικόνας, πολλές εταιρείες ενδέχεται να μην έχουν τους πόρους που μπορούν να παράγουν αποδεκτά και ακριβή αποτελέσματα.

6.1 Συμπεράσματα Έρευνας

Αρχικά αναφερθήκαμε στις νέες τεχνολογίες 5G και IoT κατά πόσο έχουν διεισδύσει στη σύγχρονη ζωή καθώς και πόσες εφαρμογές μπορούν να έχουν στο άμεσο μέλλον σχετικά με τα έξυπνα σπίτια, έξυπνα αυτοκίνητα και έξυπνες υπηρεσίες σε υγειονομικές ανάγκες. Από την ερευνά μας προκύπτει ότι οι συγκεκριμένες εφαρμογές απαιτούν μεγάλους πόρους εξυπηρέτησης , μεγάλη υπολογιστική ισχύ και γρήγορες ταχύτητες.

Επικεντρωθήκαμε στο θεωρητικό υπόβαθρο της εργασίας που αφορά τις τεχνολογίες εικονικοποίησης, Στον κλάδο των τηλεπικοινωνιών η εικονικοποίηση δικτυακών λειτουργιών είναι αρχιτεκτονική δικτύου που χρησιμοποιεί τις τεχνολογίες εικονικοποίησης για να εξομοιώσει λειτουργίες κόμβων δικτύων σε δομικά στοιχεία που μπορούν να συνδεθούν μαζί για να δημιουργήσουν υπηρεσίες τηλεπικοινωνιών. Στο edge computing όπου είναι, "ένας κατανεμητής ανοιχτής αρχιτεκτονικής IT που διαθέτει αποκεντρωμένη επεξεργαστική ισχύ, υποστηρίζοντας τις τεχνολογίες του mobile computing και του Internet of Things (IoT)". Στο Edge Computing, τα δεδομένα υποβάλλονται σε επεξεργασία από την ίδια τη συσκευή ή από έναν τοπικό υπολογιστή ή διακομιστή, αντί να μεταδίδονται σε ένα κέντρο δεδομένων».

Έγινε αναφορά στη Μηχανική Μάθηση η οποία είναι υποπεδίο της επιστήμης των υπολογιστών που αναπτύχθηκε από τη μελέτη της αναγνώρισης προτύπων και της υπολογιστικής θεωρίας μάθησης στην τεχνητή νοημοσύνη, μέσω της μηχανικής μάθησης δίνεται η δυνατότητα αναγνώρισης ανθρωπίνων λειτουργιών από συσκευές. Η ανίχνευση αντικειμένων είναι μια τεχνολογία της πληροφορικής που σχετίζεται με την όραση και την επεξεργασία των εικόνων με την ανίχνευση περιπτώσεων αντικειμένων μιας συγκεκριμένης κατηγορίας (όπως άνθρωποι, κτίρια ή αυτοκίνητα) σε ψηφιακές εικόνες και βίντεο. Πολύ καλά εξειδικευμένοι τομείς στην ανίχνευση αντικειμένων περιλαμβάνουν ανίχνευση προσώπου και ανίχνευση πεζών. Η ανίχνευση αντικειμένων έχει εφαρμογές σε πολλούς τομείς , συμπεριλαμβανομένης της ανάκτησης εικόνων και της βιντεοεπιτήρησης.

Καταγράψαμε πλατφόρμες και τεχνολογίες αναγνώρισης αντικειμένων καταλήξαμε στην πιο δημοφιλή με τη μεγαλύτερη κοινότητα χρηστών, με ολοκληρωμένη τεκμηρίωση και πληθώρα παραδειγμάτων. Εκπαιδεύσαμε ένα προ εκπαιδευμένο μοντέλο αναγνώρισης αντικειμένων κινητών τηλεφώνων. Διαπιστώθηκε ότι όσο πιο πολλά και σαφή δεδομένα

εισάγουμε στο μοντέλο τόσο μεγαλύτερο ποσοστό επίτευξης της αναγνώρισης θα πραγματοποιηθεί.

Η αυτοματοποιημένη οργάνωση εικόνων που προσφέρεται από εφαρμογές που βασίζονται σε νέφος, οι εταιρείες τηλεπικοινωνιών είναι από τους πιο δημοφιλούς αποδέκτες αυτής της τεχνολογίας που έχει βελτιώσει την εμπειρία των χρηστών .

Διάφορα οφέλη, όπως καλύτερη ασφάλεια και αυτοματοποίηση της ταυτοποίησης, είναι οι παράγοντες που ενθαρρύνουν την εφαρμογή της αναγνώρισης προσώπου σε μεγάλους δημόσιους χώρους ή εκδηλώσεις.

Η εμφάνιση πλατφορμών τεχνητής νοημοσύνης και μηχανικής μάθησης μεγάλης κλίμακας που προσφέρονται από τεχνολογικούς γίγαντες, οδήγησε στην ανάπτυξη λογισμικού επεξεργασίας εικόνων με πολλαπλές λειτουργίες όπως η αναγνώριση προσώπου και αντικειμένων και ο εντοπισμός ορόσημων.

Η αυξανόμενη ενοποίηση της πλατφόρμας επεξεργασίας ψηφιακών εικόνων και φορητών υπολογιστών σε διάφορες εφαρμογές όπως ψηφιακές αγορές και επαλήθευση εγγράφων προωθούν την ανάπτυξη της αγοράς αναγνώρισης εικόνας.

Επιλέχθηκε ένα προ εκπαιδευμένο μοντέλο για την εκπαίδευση του με τις εικόνες του πειράματος μας, υπάρχουν πληθώρα προ εκπαιδευμένων μοντέλων, χρησιμοποιήσαμε ένα που δεν εμφάνιζε προβλήματα στην αναγνώριση αντικειμένων. Η παραγωγική διαδικασία ενός μοντέλου απαιτεί το serving του σε κάποιο host ώστε να μπορεί να δεχθεί αιτήματα (requests).

Τέλος για το δικό μας μοντέλο δημιουργήθηκε κώδικας σε python για την εξομοίωση αποστολών αιτημάτων στο service για την αναγνώριση των εικόνων. Με script καταγράφονται ο χρόνος απόδοσης , η κατανάλωση μνήμης και CPU. Οι πειραματικές μετρήσεις έλαβαν χώρα σε εργαστηριακό περιβάλλον και αποδόθηκαν σε γραφήματα.

6.2 Μελλοντικές κατευθύνσεις

Ωστόσο, δε θα μπορούσαμε να μην αναφερθούμε και στη δική μας μελέτη και την περαιτέρω ανάπτυξη του πειράματος μας. Θα μπορούσε να προκύψει και κάποια μελλοντική εργασία και συμπεράσματα τα όποια θα ήταν χρήσιμα σε όλους τους συναδέλφους που θα ήθελαν να ασχοληθούν στην πράξη με το συγκεκριμένο αντικείμενο.

Από την ανάπτυξη της εργασίας και από την ευρεία πλέον χρήση της τεχνολογίας ανίχνευσης αντικειμένων συμπερασματικά μπορούμε να πούμε ότι μελλοντικά αξίζει:

1. Να επενδύσουμε στην βελτιστοποίηση των τεχνολογιών, χαρακτηριστικό που έχουν αναλάβει η εταιρείες διάθεσης τέτοιων τεχνολογιών με τις μεγάλες κοινότητες επιστημόνων.

2. Την βελτιστοποίηση των μοντέλων για πιο ακριβή αποτελέσματα και περισσότερες πληροφορίες όσων αφορά την απόδοση τους.

Για την βελτιστοποίηση του μοντέλου , υπάρχουν δυνατότητες ανάλυσης των διάφορων χρόνων του μοντέλου και διαφορετικά εργαλεία που μπορείτε να χρησιμοποιήσετε για να εντοπίσετε και να επιλύσετε σημεία συμφόρησης απόδοσης μοντέλου.

Η βελτιστοποίηση επίσης μπορεί να αναλυθεί από την εισαγωγή των δεδομένων. Η διαδικασία ανάγνωσης δεδομένων χωρίζεται σε πολλαπλά στάδια επεξεργασίας δεδομένων συνδεδεμένα σε σειρά, όπου η έξοδος ενός σταδίου είναι η είσοδος στο επόμενο. Αυτό το σύστημα ανάγνωσης δεδομένων ονομάζεται αγωγός εισαγωγής. Ένας τυπικός αγωγός για την ανάγνωση εγγραφών από αρχεία έχει τα ακόλουθα στάδια: Ανάγνωση αρχείων, Προ επεξεργασία αρχείων (προαιρετικό),Μεταφορά αρχείων από τον κεντρικό υπολογιστή στη συσκευή

Ένας αναποτελεσματικός αγωγός εισόδου μπορεί να επιβραδύνει σοβαρά την υπηρεσία και κατ' επέκταση το πρόγραμμα σας. Ένα θεωρείται δεσμευμένη στην είσοδο όταν ξοδεύει σημαντικό μέρος του χρόνου στον αγωγό εισαγωγής.

Ο αναλυτής αγωγών εισαγωγής σάς ενημερώνει αμέσως εάν το πρόγραμμά σας είναι δεσμευμένο στην είσοδο και σας καθοδηγεί στην ανάλυση συσκευών και κεντρικών υπολογιστών για να εντοπίσετε προβλήματα συμφόρησης σε οποιοδήποτε στάδιο του αγωγού εισαγωγής.

Στη διάθεση των συναδέλφων υπάρχουν εργαλεία για την αναλυτική παρακολούθηση των μοντέλων πόσο αποδίδει το μοντέλο σας στον κεντρικό υπολογιστή (CPU), στη συσκευή (GPU) ή σε συνδυασμό τόσο του κεντρικού υπολογιστή όσο και της συσκευής. Ωστόσο τα παραπάνω εργαλεία διατίθενται για εγκαταστάσεις GPU.

Ένα τέτοιο εργαλείο είναι ο profiler tensorflow βοηθά να κατανοήσετε την κατανάλωση πόρων υλικού (χρόνος και μνήμη) των διαφόρων λειτουργιών TensorFlow (ops) στο μοντέλο σας και να επιλύσετε τα σημεία συμφόρησης απόδοσης και, τελικά, να κάνετε το μοντέλο να εκτελείται πιο γρήγορα.

Παρακάτω θα δούμε τον Profiler, τα διάφορα διαθέσιμα εργαλεία, τους διαφορετικούς τρόπους με τους οποίους το Profiler συλλέγει δεδομένα απόδοσης και μερικές προτεινόμενες βέλτιστες πρακτικές για τη βελτιστοποίηση της απόδοσης του μοντέλου.

6.2.1 Εργαλεία Αξιολόγησης Μοντέλων

Υπάρχουν σουίτες εφαρμογών με πολλά διαθέσιμα εργαλεία για την βελτιστοποίηση των μοντέλων που εστιάζουν στους παρακάτω παράγοντες

- Overview page
- Input pipeline analyzer
- TensorFlow stats
- Trace viewer
- GPU kernel stats

Παρακάτω αναφέρονται λίγα λόγια για το κάθε εργαλείο καθώς και πλήρες εγχειρίδιο χρήσης θα βρείτε στην επίσημη του σελίδα[53].

Overview page

Η σελίδα επισκόπησης παρέχει μια προβολή ανώτατου επιπέδου για την απόδοση του μοντέλου κατά τη διάρκεια της εκτέλεσης ενός προφίλ. Η σελίδα σάς δείχνει μια συγκεντρωτική σελίδα επισκόπησης για τον κεντρικό υπολογιστή σας και όλες τις συσκευές, καθώς και ορισμένες προτάσεις για τη βελτίωση της απόδοσης της εκπαίδευσης του μοντέλου.

Input pipeline analyzer

Όταν ένα πρόγραμμα TensorFlow διαβάζει δεδομένα από ένα αρχείο, ξεκινά στην κορυφή του γραφήματος TensorFlow. Η διαδικασία ανάγνωσης χωρίζεται σε πολλαπλά στάδια επεξεργασίας δεδομένων συνδεδεμένα σε σειρά, όπου η έξοδος ενός σταδίου είναι η είσοδος στο επόμενο. Αυτό το σύστημα ανάγνωσης δεδομένων ονομάζεται είσοδος pipeline (input pipeline).

TensorFlow stats

Το εργαλείο TensorFlow Stats εμφανίζει την απόδοση κάθε TensorFlow op (op) που εκτελείται στον κεντρικό υπολογιστή ή τη συσκευή κατά τη διάρκεια μιας περιόδου λειτουργίας προφίλ.

Trace viewer

Το Trace viewer επιτρέπει να εντοπίσετε προβλήματα απόδοσης στο μοντέλο και, στη συνέχεια, να λάβετε μέτρα για την επίλυσή τους. Για παράδειγμα, σε υψηλό επίπεδο, μπορείτε να προσδιορίσετε εάν η εκπαίδευση του μοντέλου απαιτεί το μεγαλύτερο μέρος του χρόνου. Αναλυτικά, μπορείτε να προσδιορίσετε ποιες λειτουργίες χρειάζονται περισσότερο χρόνο εκτέλεσης.

GPU kernel stats

Αυτό το εργαλείο εμφανίζει στατιστικά στοιχεία απόδοσης και το αρχικό op για κάθε επιταχυνόμενο πυρήνα GPU.

Τέλος, κάθε μοντέλο μπορεί να βελτιωθεί είτε κάνοντας optimization στα δεδομένα εισαγωγής μια αποτελεσματική είσοδος δεδομένων μπορεί να βελτιώσει δραστικά την ταχύτητα εκτέλεσης του μοντέλου μειώνοντας τον χρόνο αδράνειας της συσκευής. Κάποιοι τρόποι θα μπορούσαν να είναι με τις τεχνικές τους ορολογίες:

- Prefetch data

[53] https://www.tensorflow.org/guide/profiler

- Parallelize data extraction
- Parallelize data transformation
- Cache data in memory
- Vectorize user-defined functions
- Reduce memory usage when applying transformations

Είτε βελτιώνοντας την απόδοση της συσκευής με τους παρακάτω τρόπους, όπου εδώ πάλι η αναφορά θα γίνει με τις τεχνικές ορολογίες:

- Increase training mini-batch size (number of training samples used per device in one iteration of the training loop)
- Use TF Stats to find out how efficiently on-device ops run
- Minimize host Python operations between steps and reduce callbacks. Calculate metrics every few steps instead of at every step
- Keep the device compute units busy
- Send data to multiple devices in parallel

Σχετικά με την απόδοση των μοντέλων και τα εργαλεία υπάρχει μια πολύ μεγάλη κοινότητα χρηστών η οποία εξελίσσει και βελτιώνει συνεχώς νέες εκδόσεις για τις εφαρμογές που προσφέρονται για την αναγνώριση αντικειμένων. Οπότε η παρακολούθηση των τεχνολογιών είναι μονόδρομος για κάποιον που θέλει να ασχοληθεί, στην παρούσα εργασία έχουν δοθεί όλες οι πηγές που λήφθηκαν τα εργαλεία που χρησιμοποιήθηκαν.

6.3 Βέλτιστες πρακτικές για βέλτιστη απόδοση μοντέλου

Μπορούν να ακολουθηθούν οι παρακάτω κατευθύνσεις οι οποίες ισχύουν για τα μοντέλα ώστε να επιτευχθεί η μέγιστη απόδοση τους.

6.3.1 Βελτιστοποίηση του τρόπου εισόδου δεδομένων στο μοντέλο

Ένας αποτελεσματικός τρόπος εισαγωγής δεδομένων μπορεί να βελτιώσει δραστικά την ταχύτητα εκτέλεσης του μοντέλου μειώνοντας τον χρόνο αδράνειας της συσκευής. Εξετάστε το ενδεχόμενο να ενσωματώσετε τις ακόλουθες βέλτιστες πρακτικές, όπως περιγράφονται λεπτομερώς εδώ, για να κάνετε την εισαγωγή δεδομένων πιο αποτελεσματικά:

- Prefetch data μεταφορά (δεδομένα) από την κύρια μνήμη σε προσωρινή αποθήκευση σε ετοιμότητα για μελλοντική χρήση.

- Parallelize data extraction- Παράλληλη εξαγωγή δεδομένων
- Parallelize data transformation- Παράλληλος μετασχηματισμός δεδομένων
- Cache data in memory- Φόρτωση δεδομένων προσωρινής μνήμης στη μνήμη
- Vectorize user-defined functions-Εφαρμογή συναρτήσεων καθορισμένων από το χρήστη
- Reduce memory usage when applying transformations- Μειώστε τη χρήση μνήμης κατά την εφαρμογή μετασχηματισμών

Επιπρόσθετα , το μοντέλο θα πρέπει να δοκιμαστεί και εκτελεστεί με πολλαπλά σύνθετα δεδομένα ως δεδομένα εισόδου για να ελεγχθεί τόσο ο τρόπος εισαγωγής των δεδομένων και αν αυτός δημιουργεί εμπόδια απόδοσης, φαινόμενο bottleneck.

6.3.2 Βελτίωση απόδοσης συσκευής

Αύξηση του μεγέθους μικρής παρτίδας εκπαίδευσης (αριθμός δειγμάτων εκπαίδευσης που χρησιμοποιήθηκαν ανά συσκευή σε μία επανάληψη του βρόχου προπόνησης)

- Χρήση στατιστικών εργαλείων για την για να μάθετε πόσο αποτελεσματικά λειτουργούν οι λειτουργίες στη συσκευή
- Χρήση συναρτήσεων νεότερων εκδόσεων του framework που χρησιμοποιεί το μοντέλο για την εκτέλεση υπολογισμών.
- Ελαχιστοποιήστε τις λειτουργίες κεντρικού υπολογιστή Python μεταξύ βημάτων και μειώστε τις επιστροφές κλήσεων
- Υπολογίστε τις μετρήσεις κάθε λίγα βήματα αντί για κάθε βήμαKeep the device compute units busy
- Αποστολή δεδομένων σε πολλές συσκευές παράλληλα

Γενικότερα η βελτίωση απόδοσης ενός μοντέλου και οι μετρήσεις οφείλουν να πραγματοποιηθούν σε ίδια λειτουργικά περιβάλλοντα με στον ίδιο edge computer τροποιώντας τα δεδομένα εισαγωγής τα οποία πρέπει να αποτελούνται από πολλά διαφορετικά σύνολα αιτημάτων τόσο σε ποιότητα όσο και σε ποσότητα και μετρήσεις των αποδόσεων θα πρέπει να γίνονται με εργαλεία του ίδιου framework. Τέλος η αξιολόγηση ενός μοντέλου θα μπορούσε ακόμη να γίνει και με δυο διαφορετικά framework ώστε να συγκριθούν και διαφορετικές τεχνολογίες εκπαίδευσης και καταγραφής μετρήσεων.

Φυσικά πλέον υπάρχουν και cloud repository όπου δίνουν τη δυνατότητα χρήσης πόρων για παρόμοιες μελέτες μοντέλων.

Παράρτημα Α-Κώδικας Script

Client Api

```
from __future__ import print_function
from PIL import Image
import cv2
import json
from tensorflow. keras. preprocessing. Image  import img_to_array, load_img
import requests
import base64
import requests
import numpy as np
import psutil
import os

SERVER_URL = 'http://server_ip:8501/v1/models/cellphone:predict'

# The image URL is the location of the image we should send to the server

def main ():
  # Download the image
  #dl_request = requests.get(IMAGE_URL, stream=True)
  #dl_request.raise_for_status()
  t=psutil.Process(6712)
  p = psutil.Process()
  p.cpu_percent(interval=None)

  image ='image4.jpg'

  # Convert arbitrary sized jpeg image to 28x28 b/w image.
  headers = {"content-type": "application/json"}
  image_content =cv2.imread(image,1).astype('uint8').tolist()
  body = {"instances":[{"inputs":cv2.imread(image,1).astype('uint8').tolist()}]}
  r = requests.post(SERVER_URL, data=json.dumps(body), headers = headers)
  print(r.text)

  # Send few actual requests and report average time.
```

```
 total_time = 0
 num_requests =[20,30,40,50,60,70,80,90,100,110,120,130,140,150,160,170,180,190,200]
 for num_r in num_requests:
  for x in range(num_r):
    response = requests.post(SERVER_URL, data=json.dumps(body), headers=headers)
    total_time += response.elapsed.total_seconds()
    avg_time=(total_time/num_r)
    pusage_cpu=p.cpu_percent(interval=None)
    pmemory=p. memory_percent ()
    tcpu=t.cpu_percent(interval=None)
    tmemory=t. memory_percent ()
  #avg_time=(total_time/num_requests)
  #prediction = response. json()['predictions'][0]

  avg_time=(total_time/num_r)
  print('num_requests:{}'.format(num_r))
  print('avg_time:{} s'.format((avg_time)))
  print('total_time:{} s'.format((total_time)))
  print()
  #p = psutil.Process()
  #print(p)
  #print("Memory",pmemory)
  #print("cpu",pusage_cpu)
  print()
  t= psutil.Process(6712)
  print(t)
  print("Memory",tmemory)
  print("cpu",tcpu)
  print()


if __name__ == '__main__':
 main()
```

Παράρτημα Β-Πηγές

[1] "What is Cloud Computing?". Amazon Web Services. 2013-03-19. Retrieved 2013-03-20

[2] Gruman, Galen (2008-04-07). "What cloud computing really means". *InfoWorld*. Retrieved 2009-06-02

[3] "What Is Cloud Computing?". *PCMAG*. Retrieved 2020-02-24.

[4] Dasiopoulou, Stamatia, et al. "Knowledge-assisted semantic video object detection." IEEE Transactions on Circuits and Systems for Video Technology 15.10 (2005): 1210–1224.

[5] https://en.wikipedia.org/wiki/Object_detection

[6] Rouse, Margaret (2019)."internet of things (IoT)". IOT Agenda. Retrieved 14 August 2019

[7] "Internet of Things Global Standards Initiative".ITU.

[8] https://data-flair.training/blogs/iot-and-machine-learning/

[9] «What is the Internet of Things, and how does it work?» . Internet of Things blog.

[10] https://www.analyticsvidhya.com/blog/2017/06/transfer-learning-the-art-of-fine-tuning-a-pre-trained-model/

[11] https://towardsdatascience.com/transfer-learning-from-pre-trained-models-f2393f124751

[12] https://docs.aws.amazon.com/machine-learning/latest/dg/training-ml-models.html

[13] https://en.wikipedia.org/wiki/Training,_validation,_and_test_sets

[14] Dataset Shift in Machine LearningFebruary 2009, The MIT Press, Neil D. Lawrence, Anton Schwaighofer, Masashi Sugiyama, Joaquin Quionero-Candela

[15] https://machinelearningmastery.com/object-recognition-with-deep-learning/

[16] Large Scale Machine Learning with Python Bastiaan Sjardin, Luca Massaron, Alberto Boschetti ,Packt Publishing Ltd

[17] https://en.wikipedia.org/wiki/Machine_learning

[18] Mitchell, Tom (1997). Machine Learning. New York: McGraw Hill

[19] Ethem Alpaydin (2020). Introduction to Machine Learning (Fourth ed.). MIT

[20] TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems (Preliminary White Paper, November 9, 2015), https://arxiv.org/pdf/1603.04467.pdf%20http://arxiv.org/abs/1603.04467.pdf

[21] Goldberg, David E.; Holland, John H. (1988). "Genetic algorithms and machine learning"

[22] https://towardsdatascience.com/tensorflow-serving-with-docker-9b9d87f89f71

[23] https://www.tensorflow.org/tfx/guide/serving

[24] https://www.tensorflow.org/tfx/serving/api_rest

[25] Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, Aurelien Geron, O'Reilly Media

[26] da Costa, CA; Pasluosta, CF; Eskofier, B; da Silva, DB; da Rosa Righi, R (July 2018). "Internet of Health Things: Toward intelligent vital signs monitoring in hospital wards". Artificial Intelligence in Medicine

[27] Engineer, A; Sternberg, EM; Najafi, B (21 August 2018). "Designing Interiors to Mitigate Physical and Cognitive Deficits Related to Aging and to Promote Longevity in Older Adults: A Review"

[28] Kricka, LJ (2019). "History of disruptions in laboratory medicine: what have we learned from predictions?"

[29] Topol, Eric (2016). The Patient Will See You Now: The Future of Medicine Is in Your Hands. Basic Books

[30] Mahmud, Khizir; Town, Graham E.; Morsalin, Sayidul; Hossain, M.J. (February 2018). "Integration of electric vehicles and management in the internet of energy". Renewable and Sustainable Energy Reviews.

[31] Xie, Xiao-Feng; Wang, Zun-Jing (2017). "Integrated in-vehicle decision support system for driving at signalized intersections: A prototype of smart IoT in transportation". Transportation Research Board (TRB) Annual Meeting, Washington, DC, USA

[32] http://www.wiomax.com/what-can-the-smart-iot-transform-transportation-and-smart-cities/

[33] Mitchell, Tom (1997). Machine Learning. New York: McGraw Hill

[34] Ethem Alpaydin (2020). Introduction to Machine Learning (Fourth ed.),MIT

[35] Garbade, Dr Michael J. (14 September 2018). https://towardsdatascience.com/clearing-the-confusion-ai-vs-machine-learning-vs-deep-learning-differences-fce69b21d5eb ". Medium. Retrieved 28 October 2020.

[36] https://www.ibm.com/cloud/blog/ai-vs-machine-learning-vs-deep-learning-vs-neural-networks, *www.ibm.com*. Retrieved 28 October 2020.

[37] http://www.journalimcms.org/special_issue/an-empirical-science-research-on-bioinformatics-in-machine-learning/, Retrieved 28 October 2020

[38] https://en.wikipedia.org/wiki/ECML_PKDD

[39] https://www.researchgate.net/publication/327386452_Machine_Learning_in_Big_Data

[40] https://en.wikipedia.org/wiki/Object_detection#Methods

[41] http://www.hbcse.tifr.res.in/jrmcont/notespart1/node45.html , Top-down vs. bottom-up approaches

[42] On Multi-Access Edge Computing: A Survey of the Emerging 5G Network Edge Cloud Architecture and Orchestration, http://www.mosaic-lab.org/uploads/papers/c191e2bf-70d4-40ed-ba6d-e82f0c4c156c.pdf

[43] Survey on Multi-Access Edge Computing for Internet of Things Realization, https://arxiv.org/pdf/1805.06695.pdf

[44] A survey of mobile cloud computing: architecture, applications, and approaches, https://onlinelibrary.wiley.com/doi/pdf/10.1002/wcm.1203

[45] Towards secure mobile cloud computing: A survey, http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.401.7388&rep=rep1&type=pf

[46] https://en.wikipedia.org/wiki/Augmented_reality

[47] https://justaskthales.com/en/difference-4g-5g/, ust Ask Gemalto EN. March 14, 2018. Retrieved January 3, 2020.

[48] http://image-net.org/challenges/LSVRC/2015/

[49] https://www.businessinsider.com/iot-infrastructure-technology?r=US&IR=T

[50] https://en.wikipedia.org/wiki/Network_function_virtualization

[51] https://www.etsi.org/technologies/nfv

[52] Network Functions Virtualisation – Introductory White Paper, October 22-24, 2012 at the “SDN and OpenFlow World Congress”, Darmstadt-Germany.(pdf)

[53] https://www.openstack.org/

[54] https://en.wikipedia.org/wiki/OpenStack

[55] https://www.openstack.org/software/roadmap/

[56] https://osm.etsi.org/wikipub/index.php/OSM_Release_FIVE

[57] https://en.wikipedia.org/wiki/Edge_computing

[58] Davis, A.; Parikh, J.; Weihl, W. (2004). "EdgeComputing: Extending Enterprise Applications to the Edge of the Internet".

[59] Hamilton, Eric (27 December 2018). "What is Edge Computing: The Network Edge Explained". cloudwards.net

[60] Nygren., E.; Sitaraman R. K.; Sun, J. (2010). "The Akamai Network: A Platform for High-Performance Internet Applications" (PDF).

[61] NIST Special Publication 800-145 The NIST Definition of Cloud Computing Peter Mell,Timothy Grance

[62] https://en.wikipedia.org/wiki/Cloud_computing

[63] https://www.gartner.com/en/information-technology/glossary/information-platform-as-a-service-ipaas

[64] https://www.gartner.com/en/documents/1729256, Model for Integration PaaS

[65] K. Balaji ME, K. Lavanya PhD, in Deep Learning and Parallel Computing Environment for Bioengineering Systems, 2019

[66] The Complete Beginner's Guide to Deep Learning: Convolutional Neural Networks and Image Classification, https://towardsdatascience.com/

[67] https://en.wikipedia.org/wiki/Contextual_image_classification

[68] https://www.tensorflow.org/

[69] https://en.wikipedia.org/wiki/OpenCV

[70] 5GINFIRE-D2-COSMOS-v1.0,pdf,

[71] https://5ginfire.eu/cosmos/

[72] 5GINFIRE-D3-COSMOS-v1.0.pdf