ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

Αποθορυβοποίηση εικόνων βάθους με χρήση εποπτευόμενης μάθησης

Depth Images Denoising with Supervised Learning

Θεόδωρος Τζιόλας

30/12/2020

ΥΠΕΥΘΥΝΟΣ ΚΑΘΗΓΗΤΗΣ: Δρ. ΝΙΚΟΛΑΟΣ ΔΗΜΗΤΡΙΟΥ

Π.Μ.Σ. «**ΕΝΕΡΓΕΙΑ ΚΑΙ ΣΥΣΤΗΜΑΤΑ ΑΥΤΟΜΑΤΙΣΜΩΝ**»

Διπλωματική εργασία στα πλαίσια του προγράμματος μεταπτυχιακών σπουδών «Ενέργεια και Συστήματα Αυτοματισμών», με θέμα την αποθορυβοποίηση εικόνων βάθους με τη χρήση εποπτευόμενης μάθησης.

Υπεύθυνη Δήλωση

Με ατομική μου ευθύνη και γνωρίζοντας τις κυρώσεις(2), που προβλέπονται από τις διατάξεις της παρ. 6 του άρθρου 22 του Ν. 1599/1986, δηλώνω ότι: Δηλώνω υπεύθυνα ότι η συγκεκριμένη μεταπτυχιακή διπλωματική εργασία για τη λήψη του μεταπτυχιακού τίτλου σπουδών του ΠΜΣ Πλήρους Φοίτησης του Πανεπιστημίου Θεσσαλίας «Ενέργεια και Συστήματα Αυτοματισμών» έχει συγγραφεί από εμένα προσωπικά και δεν έχει υποβληθεί ούτε έχει εγκριθεί στο πλαίσιο κάποιου άλλου μεταπτυχιακού ή προπτυχιακού τίτλου σπουδών, στην Ελλάδα ή στο εξωτερικό. Η εργασία αυτή έχοντας εκπονηθεί από εμένα, αντιπροσωπεύει τις προσωπικές μου απόψεις επί του θέματος και το κείμενο είναι γραμμένο με τα δικά μου λόγια και δεν αποτελεί προϊόν λογοκλοπής από τρίτες πηγές. Οι πηγές στις οποίες ανέτρεξα για την εκπόνηση της συγκεκριμένης διπλωματικής αναφέρονται στο σύνολό τους, δίνοντας πλήρεις αναφορές στους συγγραφείς, συμπεριλαμβανομένων και των πηγών που ενδεχομένως χρησιμοποιήθηκαν από το διαδίκτυο.

υπευθυνη δηλωση μεταπτυχιακού.pdf

ΠΕΡΙΛΗΨΗ

Η διπλωματική αυτή πραγματοποιήθηκε από τον σπουδαστή του Μεταπτυχιακού Προγράμματος Σπουδών «Ενέργεια και Συστήματα Αυτοματισμών» Τζιόλα Θεόδωρο στα πλαίσια του προγράμματος σπουδών. Το θέμα της εργασίας είναι «Αποθορυβοποίηση εικόνων βάθους με χρήση εποπτευόμενης μάθησης».

Μέσα από μια κάμερα RealSense μπορούμε να πάρουμε δεδομένα εικόνων χρώματος και βάθους. Τα δεδομένα αυτά, μπορούμε μετατρέποντας τα σε τρισδιάστατα αρχεία νεφών σημείων (point clouds), να τα συγκρίνουμε με ένα Laser ακριβείας μετρήσεων τύπου LIDAR (LIght Detection And Ranging), για την απεικόνιση βάθους των ίδιων αντικειμένων. Από τη σύγκριση προκύπτει ότι, τα δεδομένα της Real Sense περιλαμβάνουν περισσότερο θόρυβο στις μετρήσεις. Μέσα από αυτή τη διπλωματική, εξετάζεται η αποθορυβοποίηση των εικόνων βάθους της κάμερας RealSense με τη χρήση εποπτευόμενης μάθησης και με στόχο αποτελέσματος την ανάλυση των αρχείων εικόνων βάθους, που προέκυψαν από τα αρχεία του ConoPoint-10.

Στόχος είναι να εξεταστεί, αν αισθητήρες χαμηλότερης απόδοσης που έχουν ένα μεγάλο εύρος εφαρμογών και φθηνότερο κόστος, μπορούν να χρησιμοποιηθούν σε πιο απαιτητικές εφαρμογές, όπου το κόστος για ακριβούς αισθητήρες τύπου ConoPoint-10 είναι ασύμφορο και δεν ενδείκνυται για τέτοιες εφαρμογές (αντοχή, έκθεση σε εξωτερικές καιρικές συνθήκες, συχνότητα βλαβών και αντικατάσταση). Έτσι, μέσω ενός συνελικτικού νευρωνικού δικτύου, η έξοδος του χαμηλότερου κόστους αισθητήρα, πριν την χρησιμοποίηση από το λογισμικό, θα φιλτράρεται και θα προκύπτει υψηλότερης ακρίβειας έξοδος με καλύτερα δεδομένα απεικόνισης, κατάλληλη για πιο εξιδεικευμένες εφαρμογές.

Τα δεδομένα εικόνων, για να είναι εφικτή η εποπτευόμενη μάθηση και η βελτίωση ποιότητας (Quality Enhancement) επεξεργάζονται για να καταλήξουν στην ίδια γωνία θέασης, και από νέφη σημείων μετατρέπονται σε δισδιάστατες εικόνες βάθους (Depth Maps). Επίσης πραγματοποιούνται τεχνικές επαύξησης των δεδομένων, για περισσότερα αρχεία εκπαίδευσης. Η υλοποίηση των αλγόριθμων μηχανικής μάθησης, συνελικτικών νευρωνικών δικτύων και η προ-επεξεργασία των εικόνων πραγματοποιείται με την γλώσσα προγραμματισμού Python, με τη χρήση βιβλιοθηκών ανοιχτού λογισμικού.

Επίσης γίνεται αναφορά στη βασική θεωρία που περιλαμβάνει η τρισδιάστατη απεικόνιση και η μηχανική όραση (Machine Vision) και αναλύονται τα χρήσιμα συμπεράσματα, που προκύπτουν από τα τελικά αποτελέσματα.

Λέξεις κλειδί: Συνελικτικά νευρωνικά δίκτυα, τρισδιάστατη απεικόνιση, νέφη σημείων, εικόνες βάθους, επιβλεπόμενη μάθηση, Αλγόριθμοι βαθιάς μάθησης

ABSTRACT

This dissertation was created by the postgraduate student Tziolas Theodoros of the MSc. Program in “Energy and Automation Systems” by the General Department in University of Thessaly, on the topic of “depth images denoising with supervised learning”.

RGB and depth cameras like the RealSense camera can provide useful data with depth measurement. Most of the times, data from these sensors tend to be noisy. If we compare data from depth sensor, by converting depth images to point clouds, with a “LIDAR’s” point cloud for the same object, the comparison of these point clouds proves, that the LIDAR technique is far more accurate, in the quality and the depth measurement noise. The scope is to examine, if we can map low quality depth images to higher quality ones, from more efficient sensors with CNNs and supervised learning. The Convolutional Neural Network will take RealSense depth images as input and depth images of the same object from a ConoPoint-10 sensor as output. With this technique we can examine if the CNN can denoise the low-quality depth images and enhance the overall quality. An enhancement like this can improve camera sensors without changing the hardware components.

For the supervised learning, a pre-process of the point clouds is necessary to match the field of view. After the pre-process, point clouds are converted to 2d depth images. Different data augmentation techniques are used to enlarge the training dataset for further experiments and better results, with a variety of CNNs architectures. The Python language is used for the implementation of the machine learning algorithms and the pre-process of the point clouds, through open-source libraries.

The basic theory that concludes 3d visualization and computer vision is also explained.

Keywords: Convolutional Neural Networks, Point cloud, Supervised Learning, 3d, Depth images, Python, LIDAR, denoising

ΕΥΧΑΡΙΣΤΙΕΣ

Η διπλωματική αυτή είναι ο επίλογος σε ένα πρόγραμμα σπουδών, που διεύρυνε τους ορίζοντες μου και προσέφερε, χρήσιμα εφόδια στην επαγγελματική εξέλιξη μου, στον τομέα της Ενέργειας και των Αυτοματισμών. Το σύνολο των καθηγητών διετέλεσε άψογα το έργο του και μαζί με τους συμφοιτητές μου τους ευχαριστώ για την άψογη συνεργασία.

Ιδιαίτερα θέλω να αποδώσω ευχαριστίες στους Καθηγητές που με βοήθησαν να υλοποιήσω την εργασία, Δημητρίου Νικόλαο και Λάμπρο Λεοντάρη, καθώς επίσης και στην συντονίστρια του προγράμματος, Διδάκτωρ Ελπινίκη Παπαγεωργίου για τον επαγγελματισμό της και την συνεισφορά της, στην εισαγωγή της Τεχνητής Νοημοσύνης και της μηχανικής μάθησης, μέσα από τα μαθήματα της.

Καιρόν γνώθι.

Πιττακός ο Μυτιληναίος, 650-570 π.Χ., εκ των 7 σοφών της Αρχ. Ελλάδας

Περιεχόμενα

Ο κώδικας που υλοποιήθηκε καθώς και τα αρχεία βρίσκονται στο *github*:
https://github.com/theotziol/Depth-images-Denoising

1. Εισαγωγή

Η τεχνητή νοημοσύνη είναι η τεχνολογία αιχμής στην εποχή που διανύουμε και θα επηρεάσει σε σημαντικό βαθμό τις ζωές μας, με την εισαγωγή της στην καθημερινότητα μας. Η ανάπτυξη των υπολογιστικών συστημάτων που βασίζεται στην εξέλιξη της τεχνολογίας των επεξεργαστών, σε συνδυασμό με την πληθώρα των δεδομένων στα υπολογιστικά νέφη, έδωσαν και δίνουν περιθώριο στην εξέλιξη της τεχνητής νοημοσύνης.

Ο Τζον Μακάρθι όρισε τον τομέα της Τεχνητής Νοημοσύνης ως «επιστήμη και μεθοδολογία της δημιουργίας νοημόνων μηχανών». Ενώ σύμφωνα με την ελληνική Wikipedia: «Ο όρος τεχνητή νοημοσύνη αναφέρεται στον κλάδο της πληροφορικής, ο οποίος ασχολείται με τη σχεδίαση και την υλοποίηση υπολογιστικών συστημάτων, ικανά να μιμούνται στοιχεία της ανθρώπινης συμπεριφοράς, τα οποία υπονοούν έστω και στοιχειώδη ευφυΐα που περιλαμβάνουν έννοιες όπως:

- μάθηση,
- προσαρμοστικότητα,
- εξαγωγή συμπερασμάτων,
- κατανόηση από συμφραζόμενα,
- επίλυση προβλημάτων κλπ.»[17]

Η τεχνητή νοημοσύνη (ΤΝ) επηρεάζει και ερευνάται από πολλούς τεχνολογικούς κλάδους και έχει εφαρμογές σε πολλές επιστήμες. Και συγκεκριμένα: «αποτελεί σημείο τομής μεταξύ πολλαπλών επιστημών όπως της πληροφορικής, της ψυχολογίας, της φιλοσοφίας, της νευρολογίας, της γλωσσολογίας και της επιστήμης μηχανικών και στοχεύει στη σύνθεση ευφυούς συμπεριφοράς με στοιχεία συλλογιστικής μάθησης και προσαρμογής στο περιβάλλον, για την εφαρμογή σε ρομποτικά συστήματα και προγράμματα λογισμικού.

Η ΤΝ κατηγοριοποιείται ανάλογα τον επιστημονικό στόχο μίμησης στοιχειώδης ευφυίας, σε διαφόρους τομείς όπως: επίλυση προβλημάτων, **μηχανική και βαθιά μάθηση**, ανακάλυψη γνώσης, συστήματα γνώσης κλπ. Επίσης, επιστημονικά πεδία, όπως η **μηχανική όραση** (αντικείμενο της εργασίας), η επεξεργασία φυσικής γλώσσας (Natural language Processing) ή η ρομποτική, μπορούν να τοποθετηθούν μες στο ευρύτερο πλαίσιο της τεχνητής νοημοσύνης, ως ανεξάρτητα πεδία της.»[17]

Η ανάπτυξη τεχνολογιών ΤΝ όπως η μηχανική και η βαθιά μάθηση, είναι προσιτή σε όλους όσους θέλουν να ασχοληθούν και έχουν πρόσβαση σε ηλεκτρονικό υπολογιστή. Σε γλώσσες προγραμματισμού όπως η Python υπάρχουν βιβλιοθήκες ανοιχτού κώδικα, όπως η Tensorflow, ή η Pytorch που προσφέρουν αλγόριθμους μηχανικής μάθησης. Στους αλγόριθμους ο ενδιαφερόμενος πρέπει να δώσει πακέτα δεδομένων για εκπαίδευση και πακέτα δεδομένων για στόχο εκπαίδευσης και υλοποιεί το σύστημα που τον ενδιαφέρει. Δεδομένα για πειραματισμό μπορεί να βρει σε διάφορους ιστότοπους δωρεάν η επί πληρωμή, ή να χρησιμοποιήσει δεδομένα που προσφέρουν οι βιβλιοθήκες όπως η Tensorflow. Ορισμένες προδιαγραφές, για ομαλή λειτουργία των αλγορίθμων, πρέπει να τηρούν και οι ηλεκτρονικοί υπολογιστές. Η χρήση για παράδειγμα των GPU (Graphical Process Unit) επιταχύνει την εκτέλεση των αλγορίθμων, σε σχέση με την εκτέλεση στον επεξεργαστή, αυτό έχει ως αποτέλεσμα την ανάγκη για GPU τελευταίας τεχνολογίας. Πλέον

όμως υπηρεσίες όπως το Google Colab, που προσφέρεται από την Google, επιτρέπει την σύνταξη, εκτέλεση και αποθήκευση κώδικα μέσω του διαδικτυακού νεφους της (cloud). Υπηρεσίες σαν αυτή καθιστούν περισσότερο εύκολη από ποτέ την ενασχόληση με τεχνολογίες μηχανικής μάθησης, καθώς ένας τυπικός ηλεκτρονικός υπολογιστής και μια σύνδεση στο διαδίκτυο αρκούν. Υπάρχουν αρκετές δωρεάν πηγές που μπορεί κάποιος να εκπαιδευτεί και να ξεκινήσει τα πρώτα του βήματα στον κόσμο της μηχανικής μάθησης.

Η python είναι από τις πλέον δημοφιλέστερες γλώσσες προγραμματισμού και χρησιμοποιείται ευρέως σε προβλήματα μηχανικής και βαθιάς μάθησης. Σε αυτό έχει συμβάλει η ευκολία στην χρήση της και στην αναγνωσιμότητα του κώδικα της, που την καθιστά ιδανική για αρχάριους. Σύμφωνα με το λήμμα της Wikipedia: «Η Python είναι δυναμική γλώσσα προγραμματισμού (dynamically typed), διερμηνευόμενη (interpreted), γενικού σκοπού (general-purpose) και υψηλού επιπέδου. Ανήκει στις γλώσσες προστακτικού προγραμματισμού (imperative programming) και υποστηρίζει τόσο το διαδικαστικό (procedural programming) όσο και το αντικειμενοστραφές (object-oriented programming) προγραμματιστικό υπόδειγμα (programming paradigm).

Η Python αναπτύσσεται ως ανοιχτό λογισμικό (open source) και η διαχείρισή της γίνεται από τον μη κερδοσκοπικό οργανισμό Python Software Foundation. Ο κώδικας διανέμεται με την άδεια Python Software Foundation License»[18]. Οι πολλές βιβλιοθήκες ανοιχτού κώδικα που περιέχει από την κοινότητα του ανοιχτού λογισμικού, διευκολύνουν ιδιαίτερα αρκετές εργασίες, όπως web development, σχεδιασμό γραφικών διεπαφής χρήστη (GUI), κ.α.

1.1.Μηχανική όραση

Η Μηχανική όραση είναι η νέα τεχνολογία αιχμής, στον τομέα της επιστήμης των υπολογιστών και ανήκει στο επιστημονικό πεδίο της ΤΝ. Η ιδέα που υλοποιείται, να μπορέσουν οι υπολογιστές να προσομοιώσουν την λειτουργία της ανθρώπινης όρασης, φέρνει την επανάσταση στην ρομποτική. Ο όρος που δίνει η ελληνική Wikipedia για την μηχανική όραση είναι: «Η μηχανική όραση, υπολογιστική όραση, ή τεχνητή όραση επιχειρεί να αναπαράγει αλγοριθμικά την αίσθηση της όρασης, συνήθως σε ηλεκτρονικό υπολογιστή ή ρομπότ. Η μηχανική όραση σχετίζεται με τη θεωρία και την τεχνολογία που εμπλέκονται στη σχεδίαση και κατασκευή συστημάτων που λαμβάνουν και αναλύουν δεδομένα από ψηφιακές εικόνες. Τα δεδομένα μπορούν να είναι φωτογραφίες, βίντεο, όψεις από πολλαπλές κάμερες, πολυδιάστατες εικόνες από ιατρικό σαρωτή, εικόνες βάθους κ.α.»[19]

Από την ανθρώπινη όραση στην μηχανική όραση

Σαν άνθρωποι βασιζόμαστε ίσως στο μεγαλύτερο βαθμό, στην αίσθηση της όρασης για να καταλάβουμε τον χώρο γύρω μας, ή για να εκτελέσουμε διεργασίες. Είναι μία από τις πέντε αισθήσεις, που οι περισσότεροι άνθρωποι εφοδιαζόμαστε για την αλληλεπίδραση με το περιβάλλον. Στην τεχνολογία μας επίσης, κάθε μέρα δημιουργούμε τεράστιο όγκο ψηφιακών δεδομένων από οπτικό περιεχόμενο, όπως βίντεο, φωτογραφίες κ.α.

Για να γίνει πιο κατανοητή η μηχανική όραση, πρέπει να αναφερθεί και ο τρόπος που λειτουργεί η ανθρώπινη όραση ως μία από τις πέντε αισθήσεις, το φως και το χρώμα. «Για την όραση, όργανο αντίληψης είναι οι οφθαλμοί, ενώ το αντικείμενο της αντίληψης είναι το

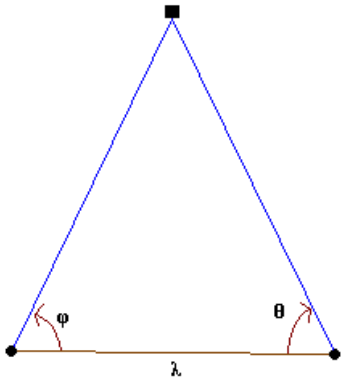
φως. Μιας και το αντικείμενο αντίληψης είναι το φως, όραση δεν υφίσταται εν απουσία αυτού. Το φως του περιβάλλοντος προσπίπτει σε διάφορα αντικείμενα και έπειτα ένα μέρος του φτάνει στα οφθαλμούς. Εκεί, οι ακτίνες προσανατολίζονται κατάλληλα, ώστε να προβληθεί στον αμφιβληστροειδή η εικόνα του περιβάλλοντος. Κατάλληλοι υποδοχείς φωτός που βρίσκονται στον αμφιβληστροειδή χιτώνα, τα κωνία και τα ραβδία, χρησιμεύουν στην αντίληψη του χρώματος και του σχήματος αντίστοιχα. Το ανθρώπινο μάτι αντιλαμβάνεται τρία χρώματα το κόκκινο, το πράσινο, το μπλε και την ένταση του φωτός στο ορατό φάσμα (Πίνακας 1-1) της ηλεκτρομαγνητικής ακτινοβολίας.» [20].

Το ορατό φως είναι ηλεκτρομαγνητική ακτινοβολία, ανήκει στο ηλεκτρομαγνητικό φάσμα και έχει συγκεκριμένες ιδιότητες κύματος (μήκος κύματος, συχνότητα, ενέργεια) *πίνακας 1-1*. «Όπου τα ηλεκτρομαγνητικά κύματα, είναι εκπομπή ηλεκτρομαγνητικής ενέργειας και χαρακτηρίζονται ως συγχρονισμένα ταλαντευόμενα ηλεκτρικά και μαγνητικά πεδία, τα οποία ταλαντώνονται σε κάθετα επίπεδα μεταξύ τους και κάθετα προς την διεύθυνση διάδοσης. Η ταχύτητα στο κενό είναι ίση με (c=299.792.458 m/s), ενώ μέσα στην ύλη η ταχύτητα είναι λίγο μικρότερη από αυτή.» [21]

Πίνακας 1-1 Ηλεκτρομαγνητικό φάσμα (Πηγή Wikipedia)

Περιοχή φάσματος	**Μήκος κύματος**	**Συχνότητες**	**Ενέργεια φωτονίων**
Ραδιοκύματα	100.000km - 1m	0-300 MHz	0 - 1,24 · 10^{-6}eV
Μικροκύματα	1m - 1mm	300 MHz - 300GHz	1,24 · 10^{-6}eV - 1,24meV
Υπέρυθρη ακτινοβολία	1mm - 740nm	300GHz - 400THz	1,24meV - 1,6eV
Ορατό φώς	740nm - 380nm	400THz-800THz	1,6 - 3,2eV
Υπεριώδης ακτινοβολία	380nm - 10nm	800THz - 3 · 10^{16}Hz	3,2eV - 124eV
Ακτίνες Χ	10nm - 0,01nm	3 · 10^{17}Hz - 3 · 10^{19}Hz	124eV - 124keV
Ακτίνες Γ	0,01nm - 0,001nm	3 · 10^{19}Hz - 3 · 10^{20}Hz	124keV - 1,24MeV
Κοσμικές ακτίνες	0,001nm - 0	3 · 10^{20}Hz -	1,24MeV -

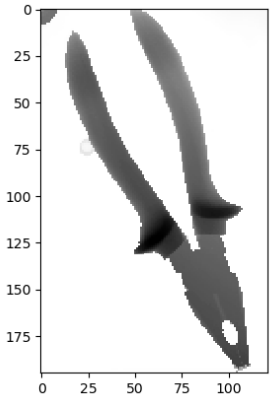
Η φυσιολογία του οπτικού συστήματος των δύο οφθαλμών δίνει στον άνθρωπο την ικανότητα να αντιλαμβάνεται την απόσταση στον τρισδιάστατο χώρο. «Υπάρχει ένα νοητό τρίγωνο που ορίζεται από τα δύο μάτια και το αντικείμενο. Γνωρίζοντας την απόσταση των ματιών και τις γωνίες που αυτά κοιτάνε το αντικείμενο μπορεί να υπολογιστεί η απόσταση του αντικειμένου. Εικόνα 1-1».[20]



Εικόνα 1-1 Μέθοδος τριγωνοποίησης [20]

Τα μάτια είναι το μέσο που χρησιμοποιεί ο εγκέφαλος βάση του φωτός να κατανοήσει το χώρο και το περιβάλλον. Τόσο τα μάτια, όσο και ο εγκέφαλος είναι απαραίτητα για ικανοποιηθούν οι συνθήκες της όρασης. Έτσι και στην μηχανική όραση, ο επεξεργαστής θα αλληλοεπιδράσει με το περιβάλλον μέσω αισθητήρων, όπως βιντεοκάμερες.

Μέσα σε μια φωτογραφία μπορούμε με σχετική άνεση να διακρίνουμε το περιεχόμενο της, δηλαδή αν υπάρχουν άνθρωποι, ζώα ή αντικείμενα, να ξεχωρίσουμε το χρώμα, την φωτεινότητα κ.α. Περιεχόμενο που στον ανθρώπινο εγκέφαλο είναι εύκολο να κατανοηθεί, αλλά στον κόσμο των bits όλα αυτά είναι αριθμοί. Για παράδειγμα στη Εικόνα 1-2 βλέπουμε μια εικόνα, όπου το αντικείμενο που απεικονίζει γίνεται εύκολα αντιληπτό.



Εικόνα 1-2 Η εικόνα μιας πένσας με διαβάθμιση του γκρί ως χρωματισμό (Greyscale). Στους x,y άξονες είναι η αρίθμηση των pixel που συνθέτουν την φωτογραφία.

Ο υπολογιστής όμως για την συγκεκριμένη εικόνα καταλαβαίνει μία δισδιάστατη συστοιχία αριθμών (2D Array), όπως φαίνεται στην Εικόνα 1-3.

```
[[138. 139. 139. ...  10.  11.  11.]
 [140. 140.   1. ...   8.   9.  10.]
 [ 15.   3.   1. ...   4.   7.   7.]
 ...
 [  0.   0.   0. ...   0.   0.   0.]
 [  0.   0.   0. ...   0.   0.   0.]
 [  0.   0.   1. ...   1.   1.   1.]]
```

Εικόνα 1-3 Η συστοιχία αριθμών (196*119 pixels), που διαβάζει ο υπολογιστής για την Εικόνα 1-2.

Ο ανθρώπινος εγκέφαλος λαμβάνει καθημερινά τεράστιο όγκο οπτικών ερεθισμάτων, τα κατανοεί, τα κατηγοριοποιεί, και λαμβάνει αποφάσεις σε μηδαμινό χρόνο. Στον κόσμο των υπολογιστών έχουμε πολύ δρόμο, για να μπορέσουμε να φτάσουμε σε ικανοποιητικό επίπεδο την λειτουργία του ανθρώπινου εγκεφάλου, πράγμα που για να υλοποιηθεί χρειάζεται τεράστια υπολογιστική ισχύ. Βέβαια η ανθρώπινη όραση μπόρεσε να εξελιχθεί μέσα σε μεγάλη χρονική περίοδο μέσω της βιολογικής εξέλιξης, σε αντίθεση με την μηχανική όραση που πρωτοεμφανίστηκε τον προηγούμενο αιώνα.

Για την δημιουργία δεδομένων μηχανικής όρασης, υπάρχουν πολλοί εξελιγμένοι αισθητήρες που χρησιμοποιούνται και επεξηγούνται παρακάτω, αλλά στη βάση όλων είναι το φαινόμενο της κάμερας μικρής οπής, (pinhole camera, camera obscura). Είναι το ίδιο φαινόμενο που συμβαίνει στο βιολογικό μάτι, όπως και στις φωτογραφικές μηχανές για την αποτύπωση της εικόνας, ενώ επίσης χρησιμοποιείται και στην ψηφιακή δημιουργία εικόνων (Computer Graphics) . Είναι ουσιαστικά η συμπεριφορά του φωτός όταν διέρχεται από μια μικρή οπή και εξετάζεται μέσω της γεωμετρίας και συγκεκριμένα με τριγωνομετρία. Αυτή η συμπεριφορά του φωτός δίνει και την ψευδαίσθηση που ονομάζεται foreshortening projection (προοπτική απεικόνιση) και αντιλαμβανόμαστε ότι τα αντικείμενα που είναι πιο κοντά σε εμάς φαίνονται μεγαλύτερα.

Ψηφιακές εικόνες , εικόνες βάθους και νέφη σημείων

Στις εικόνες (Εικόνα 1-2,Εικόνα 1-3) που παρουσιάστηκαν παραπάνω, ουσιαστικά είδαμε μια δισδιάστατη (2D) **εικόνα βάθους** και την συστοιχία που διαβάζει ο υπολογιστής για αυτήν την εικόνα. Οι εικόνες βάθους διαφέρουν από τις κανονικές εικόνες στο ότι ο χρωματισμός τους (δηλαδή η τιμή pixel της συστοιχίας), δεν αντιστοιχεί στο πραγματικό χρώμα του αντικειμένου, αλλά αντιστοιχεί σε τιμή απόστασης του πραγματικού σημείου στο χώρο, από τον αισθητήρα.

Πιο συγκεκριμένα κάθε γραμμή (row) της Εικόνα 1-3, περιέχει στήλες (columns) που δηλώνουν τα pixel του πλάτους της εικόνας, και ο συνολικός αριθμός των γραμμών δηλώνει τα pixel του ύψους της. Σε κάθε στήλη αποθηκεύεται η τιμή χρώματος του pixel (ή τιμή βάθους στον άξονα Z, για εικόνες βάθους). Στην εικόνα με διαβάθμιση του γκρί, οι τιμές είναι 8bit (0-255). Σε μια έγχρωμη εικόνα οι τιμές των pixel, είναι τιμές για κάθε χρώμα RGB (όπου R.G.B τα αρχικά στην αγγλική γλώσσα των χρωμάτων κόκκινο, πράσινο, μπλε) με συνήθως 8bit για κάθε χρώμα (jpeg και png αρχεία) και σύνολο 256*256*256 = 16,777,216 τιμές, με μερικές εικόνες κυρίως σε ανεπεξέργαστα αρχεία (Raw), να έχουν περισσότερα bit για κάθε χρώμα.

Στις δισδιάστατες εικόνες βάθους, η τιμή RGB(2^8 * 2^8 *2^8), ή Grey (2^8), δηλώνει το βάθος που έχει το pixel στην εικόνα, σε σχέση με τα άλλα pixel, σε συνάρτηση με τις πραγματικές διαστάσεις του εικονιζόμενου αντικειμένου.

Οι 2D εικόνες ενός στερεού πραγματικού αντικειμένου, είναι μια απεικόνιση σε ένα καμβά (φιλμ, αισθητήρας μηχανής).

Για τον υπολογισμό της απεικόνισης σε ένα καμβά, πρέπει να υπολογιστούν μεταβλητές όπως:

- απόσταση του καμβά από την οπή,
- το μέγεθος του καμβά,
- υπολογισμός της συμπεριφοράς του φωτός (διάθλαση, ανάκλαση σε φακούς),
- τεχνικές φωτογραφίας που χρησιμοποιούνται για τον έλεγχο της ποσότητας φως που εισέρχεται από την οπή (διάφραγμα). Τέτοιες τεχνικές στη φωτογραφία είναι, ο έλεγχος του μεγέθους διαφράγματος και η ταχύτητα του κλείστρου

Κάθε σημείο του καμβά ορίζεται στους άξονες x και y (pixel), και αντιστοιχεί σε ένα πραγματικό σημείο στον τρισδιάστατο χώρο του αντικειμένου. Στον τρισδιάστατο χώρο ένα σημείο Α ορίζεται με 3 συντεταγμένες:

A = (x,y,z) όπου x,y,z πραγματικοί αριθμοί.

Πολλές φορές για μαθηματικές πράξεις σε ένα σημείο Α δίνουμε και ένα τέταρτο πραγματικό αριθμό w που ονομάζεται ομοιογενές σημείο (homogeneous point). Έτσι:

A = (x,y,z,w)

Η μελέτη του τρισδιάστατου χώρου και της συμπεριφοράς του αντικειμένου, γίνεται μέσω της γραμμικής άλγεβρας, των διανυσμάτων, των πινάκων και της γεωμετρίας. Ένα σημείο μπορεί να ορίζεται με ένα σύστημα συντεταγμένων αναφοράς, ή με ένα δικό του σύστημα συντεταγμένων σε συνάρτηση με το σύστημα αναφοράς. [24]

Εξελιγμένες εφαρμογές κυρίως στην ρομποτική, απαιτούν την χρήση τρισδιάστατων δεδομένων. Στα τρισδιάστατα δεδομένα μας ενδιαφέρει κυρίως το βάθος , ή η απόσταση και οι πραγματικές διαστάσεις των αντικειμένων στο χώρο που λαμβάνονται για τις τρεις διαστάσεις x,y,z. Τα **νέφη σημείων** (point clouds) είναι αρχεία που χρησιμοποιούνται για την απεικόνιση τρισδιάστατων αντικειμένων και χώρου. Συνθέτουν το αντικείμενο από σημεία που ορίζονται ως μετρήσεις στους άξονες x, y, z.

Εφαρμογές του νέφους σημείων συναντάμε σε:

- Στην εξέλιξη των αυτόνομων αυτοκινήτων, που χρειάζεται διαρκή παρακολούθηση του χώρου και των εμποδίων που βρίσκονται στον δρόμο, των πεζών και της συμπεριφοράς των αυτοκινήτων.
- Στον ποιοτικός έλεγχος αντικειμένων στη βιομηχανία
- Σε ιατρικές εξετάσεις
- Στην τοπογραφία κ.α.

Αισθητήρες για την καταγραφή εικόνων βάθους και νεφών σημείων

Στην ενότητα αυτή παρουσιάζονται ενδεικτικά οι αισθητήρες για την καταγραφή τρισδιάστατων αρχείων και εικόνων βάθους, που χρησιμοποιήθηκαν για τα πειράματα της εργασίας, καθώς και ο τρόπος λειτουργίας τους.

ConoPoint-10

Είναι τρισδιάστατος αισθητήρας ολογραφίας, **υψηλής απόδοσης και κόστους** που χρησιμεύει για μέτρηση απόστασης, πάχους και περιγράμματος επιφάνειας μεταξύ 1 και 200mm.

Εικόνα 1-4 ConoPoint-10 [18]

Μερικά από τα κύρια χαρακτηριστικά του είναι:

- **Υψηλή (submicron) ακρίβεια**
- Γωνία λήψης 170^{o}
- Autoexposure
- Γραμμικός ομοαξονικός αισθητήρας κατάλληλος για μέτρηση οπών
- Ταχύτητα μέτρησης με συχνότητα laser 9000Hz

Ενώ μπορεί να χρησιμοποιηθεί σε πολλές απαιτητικές εφαρμογές όπως έλεγχος προπέλας, ελαστικών, ηλεκτροκόλλησης κ.α. [30]

Τα δεδομένα νέφη σημείων υψηλής ποιότητας που χρησιμοποιήθηκαν για την εργασία λήφθηκαν από έναν αισθητήρα Conopoint-10, με την δημιουργία ενός συστήματος σκαναρίσματος (scanning system) που υλοποιήθηκε στην έρευνα [6]. Το σύστημα περιλαμβάνει έναν ελεγκτή (controller) κίνησης Newport XPS-RL2, που ελέγχει δυο πλατφόρμες με γραμμική κίνηση FMS200CC και FM300CC, για να σκανάρει σε απόσταση 200mm και 300mm αντίστοιχα, στους άξονες x/y και να καταγράφει την απόσταση του αντικειμένου στον άξονα z μέσω του Conopoint-10.

Για την ενεργοποίηση (triggering) σε κάθε βήμα, αναπτύχθηκε ένας μηχανισμός που βασίζεται σε παλμούς στην έξοδο του controller κατά τη σύγκριση των θέσεων των πλατφόρμων, με στόχο τη μείωση του χρόνου σκαναρίσματος, χωρίς μείωση σε ακρίβεια μετρήσεων. Για ένα ολοκληρωμένο σκανάρισμα ενός αντικειμένου τυπικών διαστάσεων (για διαστάσεις αντικείμενων που χρησιμοποιήθηκαν στα δεδομένα της εργασίας), ο χρόνος που απαιτείται από την διάταξη είναι 20 λεπτά.

LIDAR

Για την μέτρηση της απόστασης υπάρχουν πολλές τεχνικές , αλλά πολλές φορές δεν είναι αρκετή η απλή μέτρηση απλά της απόστασης, γι’ αυτό και θέλουμε μια πλήρη απεικόνιση των διαστάσεων του αντικειμένου η του χώρου που εξετάζεται.

Η τεχνική LIDAR (Light Detection And Ranging) βασίζεται στην εκπομπή παλμικής ακτινοβολίας λέιζερ προς μια κατεύθυνση και ανιχνεύει το αντανακλώμενο λέιζερ και την διάρκεια που έκανε για να επιστρέψει (Time of Flight). Ο υπολογισμός της απόστασης δίνεται από τον τύπο:

$$d = c * \frac{t}{2}$$

Όπου:

d, η απόσταση

c, η ταχύτητα του φωτός ≈299.792.458 μέτρα το δευτερόλεπτο

t, ο χρόνος

Η συχνότητα των εκπομπών των παλμών, οι διαφορές στους χρόνους επιστροφής και του μήκους κύματος, χρησιμοποιούνται από τον αισθητήρα για τη δημιουργία τρισδιάστατης απεικόνισης του χώρου. Τα αρχεία που παράγουν είναι νέφη σημείων με τις συντεταγμένες x,y,z.

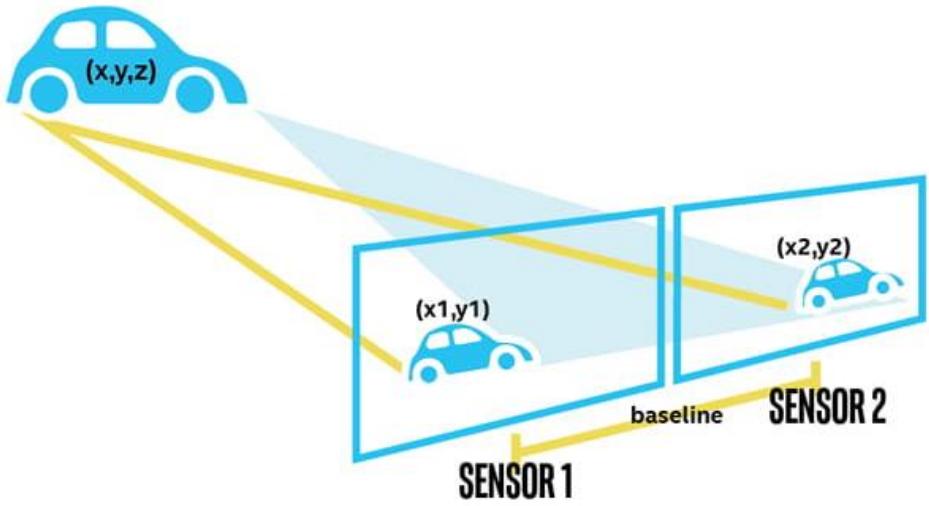
Οι αισθητήρες αυτού του είδους έχουν σχετικά ακριβό κόστος, αλλά μεγάλη έρευνα γίνεται για την ανάπτυξή τους καθώς είναι κύριο εξάρτημα στην ανάπτυξη αυτόνομων οχημάτων. Πλέον με την εισαγωγή τους στα έξυπνα τηλέφωνα, με πρώτη την εταιρεία Apple που το εγκατέστησε στα μοντέλα της, γίνεται πιο προσιτή και σε μεγαλύτερο μέρος του καταναλωτικού κοινού. Η λειτουργία του είναι ίδια με τη λειτουργία του RADAR (Radio detection and Ranging), με τα πλεονεκτήματα να είναι κυρίως στην ακρίβεια του εικονιζόμενου αντικειμένου.

Stereo Depth κάμερα RealSense D435

Η λειτουργία τους για την μέτρηση βάθους βασίζεται στην βιολογική όραση (δύο οφθαλμών), όπου τα αντικείμενα που βρίσκονται κοντά στα μάτια φαίνεται να κινούνται ταχύτερα από αντικείμενα σε μακρινή απόσταση. Έχουν δύο αισθητήρες σε απόσταση μεταξύ τους και παίρνουν δύο διαφορετικές φωτογραφίες και τις συγκρίνουν (Εικόνα 1-5). Πολλές φορές εκπέμπουν και υπέρυθρο φως για πιο ακριβή δεδομένα σε σκοτεινό περιβάλλον. Τα πλεονεκτήματα χρησιμοποιώντας αυτές τι κάμερες για εφαρμογές βάθους είναι:

- Μπορούν να χρησιμοποιηθούν πολλές κάμερες στον ίδιο χώρο. Σε αντίθεση με αισθητήρες όπως για παράδειγμα LIDAR που μπορεί να διαβάσει θόρυβο αν υπάρχει κάμερα ίδιας τεχνολογίας στον χώρο.
- Λειτουργούν σε διάφορες συνθήκες φωτισμού.
- Φθηνότερο κόστος

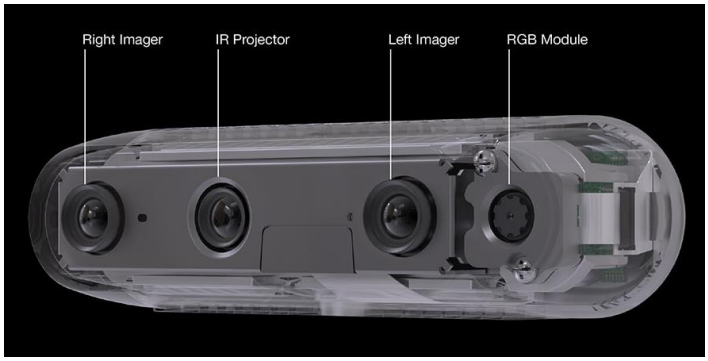
Τα αρχεία που δημιουργούν είναι εικόνες βάθους συνήθως RGBD (red, green, blue, depth) όπου πέρα από τις τιμές RGB έχουν και τιμή που αντιστοιχεί στο βάθος της φωτογραφίας. Μέσω λογισμικού μπορούν να δημιουργήσουν νέφη σημείων από τις εικόνες βάθους.



Εικόνα 1-5 Η λειτουργία των καμερών stereo depth [25]

Η απόσταση που μπορούν αυτές οι κάμερες να μετρήσουν είναι ανάλογη με την απόσταση που έχουν μεταξύ τους οι αισθητήρες, με την μέθοδο τριγωνοποίησης που αναφέρθηκε στο προηγούμενο κεφάλαιο.

Στην εργασία, για τα δεδομένα εικόνων βάθους χρησιμοποιήθηκε η κάμερα RealSense D435 που φαίνεται και στην Εικόνα 1-6.



Εικόνα 1-6 Η εικόνα της RealSense D435 και οι αισθητήρες που χρησιμοποιεί [38]

Το κόστος της είναι προσιτό σε σχέση με το ConoPoint-10 , μπορεί να χρησιμοποιηθεί σε εσωτερικούς και εξωτερικούς χώρους, ενώ και τα κύρια χαρακτηριστικά που προσφέρει:

- Ιδανική απόσταση μέτρησης από 0.3m έως 3m
- Αισθητήρας (Global Shutter) με μέγεθος pixel 3μm x 3μm
- Τεχνολογία βάθους (Active IR Stereo),
    - με οπτικό πεδίο θέασης (field of view), 86º x 57º (±3º), Ελάχιστη απόσταση μέτρησης βάθους 28cm , Ακρίβεια βάθους: <2 % στα 2m
    - Ανάλυση εικόνας βάθους 1280 x 720, ρυθμό καρέ βάθους 90 fps
- Τεχνολογία RGB, με rolling shutter αισθητήρα 2 MP
    - Ανάλυση εικόνας 1920x1080, 30 fps
    - οπτικό πεδίο θέασης (field of view), 64° × 41° × 77° (± 3°)

1.2.Μηχανική και βαθιά μάθηση

Κατά την μηχανική μάθηση ένα πρόγραμμα μαθαίνει από τα δεδομένα του, χωρίς να έχει προγραμματιστεί ρητά (Άρθουρ Σάμουελ, 1959). Ο ορισμός που αναφέρεται στο σύγγραμμα «Τεχνητή νοημοσύνη» της Γεωργούλη Α, για την μηχανική μάθηση είναι: «το φαινόμενο κατά το οποίο ένα σύστημα βελτιώνει την απόδοσή του κατά την εκτέλεση μιας συγκεκριμένης εργασίας, χωρίς να υπάρχει ανάγκη να προγραμματιστεί εκ νέου»[2].

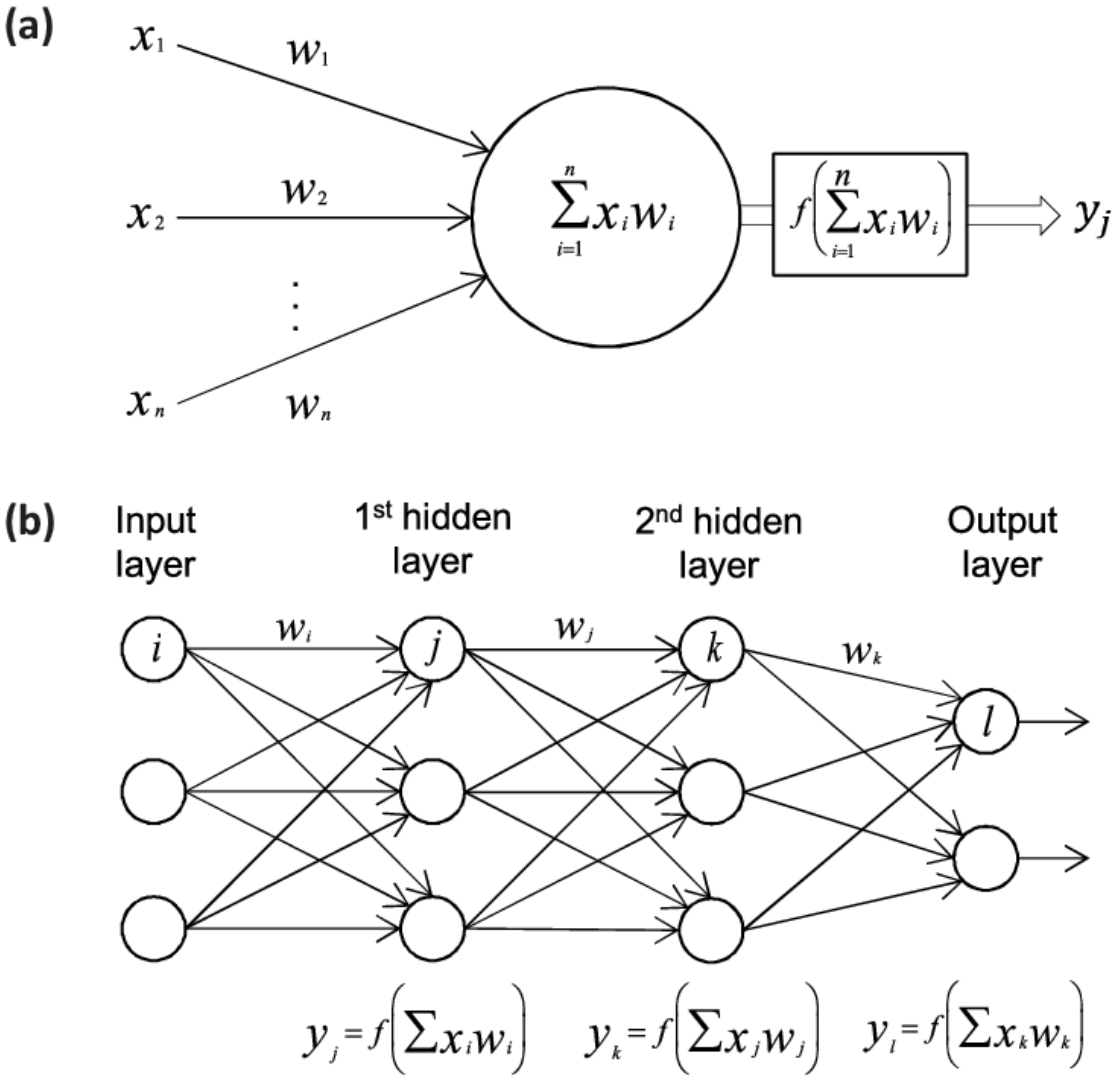
«Η Μηχανική μάθηση ανήκει στον τομέα της Τεχνητής Νοημοσύνης και αναπτύχθηκε από τη μελέτη της αναγνώρισης προτύπων και της υπολογιστικής θεωρίας μάθησης. Στη μηχανική μάθηση διερευνάται η μελέτη και η κατασκευή αλγορίθμων που μπορούν να μαθαίνουν από τα δεδομένα και να κάνουν προβλέψεις σχετικά με αυτά, ή να εξάγουν αποφάσεις, ως το αποτέλεσμα των δεδομένων.

Η μηχανική μάθηση είναι στενά συνδεδεμένη και συχνά συγχέεται με υπολογιστική στατιστική, ενώ επίσης έχει ισχυρούς δεσμούς με την μαθηματική βελτιστοποίηση.»[26]

Η βαθιά μάθηση έχει ως στόχο να λύσει την δυσκολία σε πραγματικές εφαρμογές τεχνητής νοημοσύνης, όπου τα δεδομένα μπορούν να διαφοροποιηθούν από πολλούς παράγοντες. Για παράδειγμα, αν θέλουμε να φωτογραφήσουμε ένα αντικείμενο κόκκινου χρώματος σε συνθήκες χαμηλού φωτισμού , ή σε διαφορετικού χρώματος διάχυτο φωτισμό, θα έχει ως αποτέλεσμα να αλλοιωθεί ο χρωματισμός που θα καταγράψει ο αισθητήρας. Επίσης διαφορετική γωνία λήψης φωτογραφίας ενός αντικειμένου, μπορεί να διαφοροποιεί το σχήμα του. Στην μηχανική μάθηση, σε αντίθεση με τη βαθιά μάθηση, είναι απαραίτητο να υπάρξει προ-επεξεργασία και φιλτράρισμα στα δεδομένα. Πράγμα που πολλές φορές είναι δύσκολο, ειδικά σε μεγάλο όγκο δεδομένων. Η βαθιά μάθηση καθιστά ικανό τον υπολογιστή να δημιουργήσει σύνθετες έννοιες, μέσα από απλές έννοιες.

Χαρακτηριστικό παράδειγμα ενός μοντέλου βαθιάς μάθησης είναι ένα πρόσθιας τροφοδότησης , βαθύ τεχνητό δίκτυο ή πολύ-επίπεδων νευρώνων (multilayer perceptron, ή εν συντομία MLP). Ένα τέτοιο δίκτυο είναι μια μαθηματική συνάρτηση που αντιστοιχεί ένα σύνολο δεδομένων εισόδου σε ένα σύνολο δεδομένων εξόδου.[5].

Πιο συγκεκριμένα τα τεχνητά νευρωνικά δίκτυα περιέχουν επίπεδα-στρώματα (layers), από τεχνητούς νευρώνες που προσομοιάζουν την λειτουργία του βιολογικού νευρώνα. Δέχονται δεδομένα εισόδου συνεχείς μεταβλητές x_i , τα οποία μεταβάλλονται από μία τιμή θετική ή αρνητική βάρους w_i. Μερικές φορές θεωρούμε ότι εκτός από τα σήματα και τα αντίστοιχα βάρη, ο νευρώνας έχει και κάποιο βάρος w_0(πόλωση). Στο σώμα βρίσκεται ο αθροιστής που προσθέτει τα επηρεασμένα από τα βάρη σήματα και παράγει την ποσότητα S, όπου μέσα από την μη-γραμμική συνάρτηση ενεργοποίησης $y = f(S)$ δίνεται η αριθμητική τιμή y στην έξοδο (Εικόνα 1-7). Η έξοδος είναι είσοδος άλλων νευρώνων (MLP).



Εικόνα 1-7 Η λειτουργία του νευρώνα (a) και των πολύ-επίπεδων νευρωνικών δικτύων (b).[8]

Μάθηση με επίβλεψη

Μάθηση με επίβλεψη είναι ένας από τους τρείς τρόπους μάθησης στα τεχνητά νευρωνικά δίκτυα (ΤΝΔ) και έχει σχέση με τον τρόπο που γίνεται η τροποποίηση των βαρών του. Στη μάθηση με επίβλεψη (supervised learning) δίνονται στο δίκτυο ζευγάρια διανυσμάτων εισόδου-επιθυμητής εξόδου και αυτό παράγει με την τρέχουσα κατάσταση βαρών μια έξοδο που αρχικά διαφέρει από την επιθυμητή. Αυτή η διαφορά ονομάζεται σφάλμα (error) και βάση αυτής καθώς και ενός αλγόριθμου εκπαίδευσης γίνεται συνήθως η αναπροσαρμογή των βαρών. Αλγόριθμοι βελτιστοποίησης (optimization algorithms) χρησιμοποιούνται στις εφαρμογές ΤΝΔ για την αξιολόγηση της μετατροπής των βαρών. Στον αλγόριθμο ανάστροφης μετάδοσης λάθους (Backpropagation) η μεταβολή των βαρών βασίζεται στον υπολογισμό της συνεισφοράς κάθε βάρους στο συνολικό σφάλμα.

Για τον περιορισμό καταστάσεων ατελούς μάθησης ή υπο-προσαρμογής χρειάζεται ικανοποιητικός αριθμός δεδομένων εκπαίδευσης.

Ο συνηθέστερος τρόπος χρήσης των δεδομένων εκπαίδευσης είναι σε κύκλους εκπαίδευσης που ονομάζονται εποχές (epochs).

Η εκπαίδευση τερματίζεται όταν το κριτήριο ελέγχου της ποιότητας του δικτύου, φτάσει σε κάποια επιθυμητή τιμή. Ως τέτοιο κριτήριο λαμβάνεται συνήθως το μέσο σφάλμα, ή η μεταβολή του μέσου σφάλματος του συνόλου εκπαίδευσης, που και στις δύο περιπτώσεις πρέπει να περιοριστεί σε χαμηλή τιμή.[1]

Convolutional Neural Networks

Τα συνελικτικά νευρωνικά δίκτυα (Convolutional Neural Networks ή εν συντομία CNNs) χρησιμοποιούνται σε εφαρμογές βαθιάς μάθησης που ασχολούνται και με την μηχανική όραση. Διαφέρουν από τα πλήρως συνδεδεμένα νευρωνικά δίκτυα (MLP) , καθώς δεν είναι πλήρως συνδεδεμένα και αναπτύσσουν συγκεκριμένες συνδέσεις σε κάθε στρώμα, ικανά να αναγνωρίζουν πρότυπα σε μια εικόνα, όπως για παράδειγμα, κάθετες και οριζόντιες γραμμές. Ενώ μέσα από απλά πρότυπα, μπορούν και συνθέτουν δύσκολες έννοιες. [27]

Επίσης λόγω της μη πλήρους συνδεσιμότητας, αποφεύγεται ο όρος που στην μηχανική μάθηση και τη στατιστική, ονομάζεται υπερβολική τοποθέτηση (overfitting), δηλαδή κατά την εκπαίδευση ένα νευρωνικό δίκτυο να αντιστοιχεί πλήρως τα δεδομένα εισόδου-εξόδου και να αποτυγχάνει να προβλέψει σωστά την έξοδο σε καινούρια δεδομένα εισόδου. Το υπολογιστικό κόστος επίσης είναι μικρότερο σε σχέση με τα MLP.

Τα κρυμμένα στρώματα σε ένα δίκτυο CNN είναι φίλτρα (filters), βάρη που προσαρμόζονται αυτόματα κατά την εκπαίδευση με ίδιο βάθος καναλιών (input channels), με έξοδο το εσωτερικό γινόμενο (dot product) εισόδου-εξόδου. Μια συνάρτηση μη-γραμμικότητας όπως η ReLU εφαρμόζεται στο γινόμενο. Επιπλέον , στρώματα εξαγωγής τιμών (Pooling), μέσων, μεγίστων, ελαχίστων τιμών, (max-pooling, average pooling, min-pooling) και MLP χρησιμοποιούνται σαν στρώματα σε ένα CNN.

Σημαντικές έννοιες και η επεξήγηση τους στα στρώματα των CNN είναι:

- Padding, Η διαφορά μεγέθους που προκύπτει στον αριθμό pixel με την συνέλιξη στην έξοδο, από το εσωτερικό γινόμενο μεταξύ της εισόδου και των διαστάσεων του φίλτρου (όπου συνήθως είναι μικρότερων διαστάσεων), δίνεται από τον τύπο:

  $$H_{out}\,, W_{out} = (n_h - k_h + 1) * (n_w - k_w + 1)$$

  Όπου

  n_h, n_wοι διαστάσεις στην είσοδο (μήκος, πλάτος)
  k_h, k_wοι διαστάσεις στην είσοδο (μήκος, πλάτος)

  Σε περίπτωση που δεν είναι επιθυμητή έξοδος μικρότερων διαστάσεων, εφαρμόζεται η τεχνική Padding αυξάνοντας τις διαστάσεις στην είσοδο με μηδενικά pixel. Τότε ο τύπος των διαστάσεων εξόδου είναι:

  $$H_{out}\,, W_{out} = (n_h - k_h + p_h + 1) * (n_w - k_w + p_w + 1)$$

  Με p_h, p_wτα pixel που προστέθηκαν. Για την διευκόλυνση του Padding οι διαστάσεις των φίλτρων επιλέγονται σε μονός αριθμός.

- Stride, Κατά την εφαρμογή φίλτρων το εσωτερικό γινόμενο υπολογίζεται με μετάθεση μιας θέσης pixel (Stride = 1) κατά μήκος του οριζόντιου άξονα για το πλάτος και μία θέση κατά μήκος του κάθετου άξονα για το ύψος της εικόνας.

Μπορεί να χρησιμοποιηθεί μεγαλύτερο βήμα (stride) ο τύπος που είδαμε παραπάνω αλλάζει σε:

$$H_{out}\,, W_{out} = \left[{(n_h - k_h + p_h + s_h)}/{s_h}\right] * \left[{(n_w - k_w + p_w + s_w)}/{s_w}\right]$$

Όπου s_h, s_w το stride στο ύψος και στο πλάτος αντίστοιχα.

1.3.CNN και επαύξηση δεδομένων

Τα συνελικτικά δίκτυα (CNN) όπως και όλες οι εφαρμογές μηχανικής και βαθιάς μάθησης χρειάζονται αρκετά δεδομένα εκπαίδευσης για την υλοποίηση λειτουργικών αλγορίθμων. Στην εργασία επιλέχθηκε η αύξηση του πακέτου δεδομένων (dataset), με τον γνωστό όρο στα CNNs για επαύξηση δεδομένων (Data Augmentation).

Στα CNN υπάρχουν πολλές τεχνικές για την αύξηση των δεδομένων εικόνων που προκύπτουν από την επεξεργασία των ίδιων των εικόνων, στον χρωματισμό, τον φωτισμό, τις διαστάσεις και την θέση του αντικειμένου της εικόνας. Πολλές βιβλιοθήκες επεξεργασίας εικόνων και στην python όπως η OpenCV, προσφέρουν τεχνικές και φίλτρα για την επεξεργασία εικόνων. Επίσης υπάρχουν και πολλά προγράμματα επεξεργασίας εικόνων που βοηθάνε στην ανάπτυξη του υπάρχοντος πακέτου δεδομένων, όπως το Photoshop της Adobe. Τέτοια προγράμματα δίνουν ένα περιβάλλον διεπαφής χρήστη όπου από εκεί μπορεί να προσαρμόσει την αντίθεση στον φωτισμό, των κορεσμό των χρωμάτων, την περιστροφή και την περικοπή της εικόνας.

Η επαύξηση δεδομένων κατηγοριοποιείται επίσης σε online και offline.Ο όρος Online σημαίνει ότι, οι εικόνες θα δημιουργηθούν κατά την εκτέλεση του προγράμματος και πριν την είσοδο στο νευρωνικό, ενώ offline όταν έχει γίνει προ-επεξεργασία και έχουν αποθηκευτεί σαν νέο σετ δεδομένων.

Σε εικόνες RGB, ο κορεσμός των χρωμάτων είναι μια τεχνική που μπορεί να αυξήσει το dataset και να χρησιμεύσει για δεδομένα διαφορετικών καταστάσεων φωτισμού. Τα χρώματα χωρίζονται σε ψυχρά και θερμά όπως και το φως στην τέχνη της φωτογραφίας. Διαφορετικές καιρικές συνθήκες δίνουν διαφορετικά αποτελέσματα σε μια φωτογραφία. Για παράδειγμα, μια φωτογραφία ενός αυτοκινήτου με υψηλή αντίθεση και έντονο θερμό χρώμα, έχει τραβηχτεί πιθανόν μεσημεριανές ώρες με πολύ ήλιο. Ένα CNN για κατηγοριοποίηση (classification) άμα αποτελείται μόνο από εικόνες αυτοκινήτων με θερμό φωτισμό, δεν θα μπορέσει με μεγάλη επιτυχία να καταλάβει ένα αυτοκίνητο μια συννεφιασμένη μέρα, ή ένα σούρουπο. Εφαρμόζοντας φίλτρα στις τιμές των χρωμάτων και της αντίθεσης στις φωτογραφίες εκπαίδευσης, μπορούμε να επιτύχουμε καλύτερα αποτελέσματα στο παραπάνω παράδειγμα.

Περικοπή (Crop), ή τυχαία περικοπή (Random Crop), ορίζεται η τεχνική περικοπής της αρχικής εικόνας σε μικρότερες, ώστε να δημιουργηθεί μεγαλύτερο dataset. Γίνεται είτε online, είτε offline και μπορεί να χρησιμοποιηθεί είτε με overlap (οι περικομμένες εικόνες να μοιράζονται ίδια pixel), με μικρό stride είτε χωρίς overlap ανάμεσα στα καινούρια κομμάτια εικόνας.

1.4.Αποθορυβοποίηση με χρήση εποπτευόμενης μάθησης

Στην αποθορυβοποίηση εικόνων με εποπτευόμενη μάθηση, χρησιμοποιούνται CNN ώστε να βελτιωθεί η ποιότητα εικόνων χαμηλότερης ποιότητας, με αντιστοίχιση σε εικόνες υψηλότερης ποιότητας.

Κατά την λήψη μιας εικόνας συνήθως προστίθεται θόρυβος στα δεδομένα λόγω των περιορισμών του υλισμικού (hardware) των αισθητήρων, ή εξωτερικών συνθηκών όπως για παράδειγμα του φωτισμού, ή διαφόρων άλλων παραγόντων. Ο θόρυβος μπορεί να επηρεάσει αρνητικά το μεγαλύτερο μέρος της εικόνας και να παραμορφωθούν σημαντικές λεπτομέρειες αυτής.

Τεχνικές επεξεργασίας εικόνας με φίλτρα, που ουσιαστικά είναι προκαθορισμένοι πίνακες (όπως ορίζονται στην γραμμική άλγεβρα), διαμορφώνουν τα pixel και δίνουν εφέ όπως κορεσμό χρωμάτων, αντίθεση, θολότητα, δομή και ευκρίνεια στις εικόνες και μπορούν να χρησιμοποιηθούν για να βελτιώσουν την ποιότητα της εικόνας (anisotropic diffusion, low-pass filter,non-linear filters, linear-smoothing filters). Πολλές φορές χρησιμοποιούνται και άλλες μέθοδοι για την αποθορυβοποίηση και έχουν να κάνουν είτε με στατιστικές μεθόδους (non-local means), είτε με άλλες τεχνικές όπως (Wavelet transform)[35]. Στις παρακάτω εικόνες φαίνεται για παράδειγμα, η εφαρμογή φίλτρων σε μια φωτογραφία και τα αποτελέσματα αυτών. Τέτοια φίλτρα μπορούν να βελτιώσουν εν μέρη κάποια στοιχεία της εικόνας αλλά μπορεί να προκαλέσουν και αρνητικά αποτελέσματα ταυτόχρονα.

Εικόνα 1-8 Από αριστερά προς τα δεξιά, η αρχική εικόνα, η ίδια εικόνα (μεσαία) με την χρήση φίλτρου οξύτητας (sharpen) και δεξιά με την χρήση φίλτρου θολότητας (blur). Πηγή εικόνας images.nasa.gov.

Στην παραπάνω εικόνα του γαλαξία εφαρμόστηκαν φίλτρα SHARPEN και BLUR από την βιβλιοθήκη PIL (Python Image Library).

- Το φίλτρο SHARPEN (οξύτητας), όπως δηλώνει και η ονομασία του, τροποποιεί την εικόνα κάνοντας πιο αιχμηρές τις άκρες των αντικειμένων που απεικονίζονται. Όπως φαίνεται και στην Εικόνα 1-8, λόγο της ιδιομορφίας του γαλαξία που αποτελείται από πολλά μικρά άστρα, έκανε πιο «αιχμηρό» και τον θόρυβο που περιλαμβάνει μια τέτοια φωτογραφία. Το φίλτρο αυτό δοκιμάστηκε και στην εργασία ώστε να βελτιώσει το περίγραμμα των αντικειμένων. Ο πίνακας που χρησιμοποιεί το φίλτρο sharpening της βιβλιοθήκης PIL είναι διαστάσεων 3x3 με τιμές:

-2	-2	-2
-2	32	-2
-2	-2	-2

- Το φίλτρο BLUR (θολότητας), έκανε ακριβώς το αντίθετο από το φίλτρο SHARPEN στην εικόνα και μείωσε τις οξύτητα. Αυτό το φίλτρο δεν δοκιμάστηκε στην εργασία καθώς στόχος ήταν να αυξηθεί η οξύτητα του αντικειμένου χαμηλής ποιότητας. Ο πίνακας που χρησιμοποιεί το φίλτρο BLUR της βιβλιοθήκης PIL είναι διαστάσεων 5x5 με τιμές:

1	1	1	1	1
1	0	0	0	1
1	0	0	0	1
1	0	0	0	1
1	1	1	1	1

Όταν για ένα αντικείμενο, μπορούμε να έχουμε δύο εικόνες που χαρακτηρίζονται από διαφορετική ποιότητα και προκύπτουν είτε από διαφορετικούς αισθητήρες, είτε μέσα από επεξεργασία της αρχικής εικόνας, μπορούμε να προσδιορίσουμε την διαφορά στο θόρυβο ανάμεσα τους, μέσα από αλγόριθμους και μαθηματικούς τύπους. Σε προβλήματα τέτοιου τύπου η εικόνα με καλύτερη ποιότητα ορίζεται ως βασική (Ground Truth), και βάση αυτής, οι κυριότεροι δείκτες διαφοράς ποιότητας ανάμεσα στην βασική και στην χαμηλής ποιότητας εικόνα που χρησιμοποιούνται είναι, το peak signal to noise ratio (PSNR)[15][39] και το Structure Similarity Index Measure (SSIM) [10].

- Το PSNR χρησιμοποιείται ευρέως για αξιολόγηση αποτελεσμάτων στην ποιότητα των εικόνων και ορίζεται ως το σφάλμα μέσων τετραγώνων (mean square error η MSE) ανάμεσα στις τιμές pixel των δύο εικόνων.
  Η τιμή psnr σε decibel (db) δίνεται από τον τύπο:

$$PSNR = 20 * \ log_{10}(MAX_p) - 10 * \ log_{10}(MSE)$$

  Οπού: MAX_p Η μέγιστη τιμή Pixel
  Με κανονικοποίηση (normalize), τα pixel από τιμές 0-255 λαμβάνονται ως 0 έως 1.0, που σημαίνει MAX_p= 1. οπότε

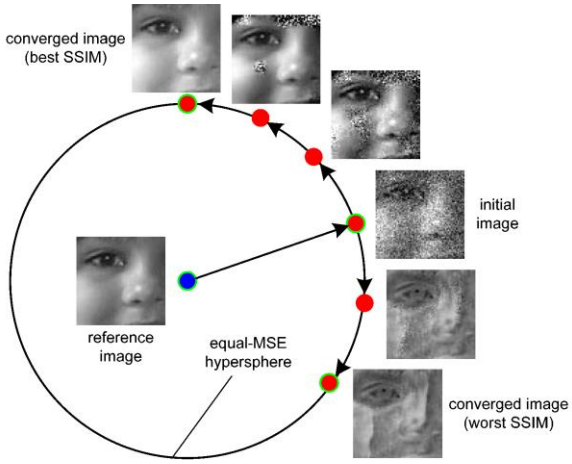
$$20 * log_{10}(1) = 0$$

  Η συνάρτηση σφάλματος μέσων τετραγώνων (MSE) ανάμεσα στην θορυβώδη μονόχρωμη εικόνα (N) και στην εικόνα Ground Truth (G) με διαστάσεις $h * w$ ορίζεται:

$$MSE = \frac{1}{h * w} \sum_{i=0}^{h-1} \sum_{j=0}^{w-1} [G(i,j) - N(i,j)]^2$$

  Το MSE είναι αντιστρόφως ανάλογο με το $PSNR$ οπότε για μικρό σφάλμα θορύβου, προκύπτει μεγάλη τιμή $PSNR$.

- Η μέθοδος SSIM χρησιμοποιείται για την μέτρηση της ομοιότητας μεταξύ δύο εικόνων, με μέτρηση πλήρης αναφοράς (full reference). Βάση αυτού είναι πιο αξιόπιστη για την σύγκριση θορύβου ανάμεσα σε δύο εικόνες, σε σχέση με το απόλυτο σφάλμα MSE (Εικόνα 1-9).



Εικόνα 1-9 Σύγκριση εικόνων με ίδιο MSE και διαφορετικό SSIM [34]

Ο αλγόριθμος SSIM πραγματοποιείται σε πολλά κομμάτια της εικόνας, όπου ο μαθηματικός τύπος μέτρησης για δύο κομμάτια x, y μεγέθους $n * n$ είναι:

$$SSIM(x,y) = \frac{(2\mu_x \mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

Όπου:

μ_x η μέση τιμή του x
μ_y η μέση τιμή του y
σ_x^2 η διακύμανση[41] του x
σ_y^2 η διακύμανση του y
σ_{xy} η συν διακύμανση του x, y
$c_1 = (k_1 L)^2$, $c_2 = (k_2 L)^2$ δύο μεταβλητές για σταθεροποίηση της διαίρεσης με μικρό παρανομαστή.
L το δυναμικό εύρος των τιμών pixel
$k_1 = 0.01$, $k_2 = 0.02$

Ο μαθηματικός τύπος βασίζεται σε σύγκριση των μετρήσεων μεταξύ των κομματιών (x, y) σε φωτεινότητα (l) , αντίθεση (c), και δομή (s)

$$l(x,y) = \frac{2\mu_x \mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1}$$

$$c(x,y) = \frac{2\sigma_{xy} + c_2}{\sigma_x^2 + \sigma_y^2 + c_2}$$

$$s(x,y) = \frac{\sigma_{xy} + c_3}{\sigma_x \sigma_y + c_3}$$

- Όπου $c_3 = c_2/2$

Και προκύπτει ο συνδυασμός με βάρη α, β, γ

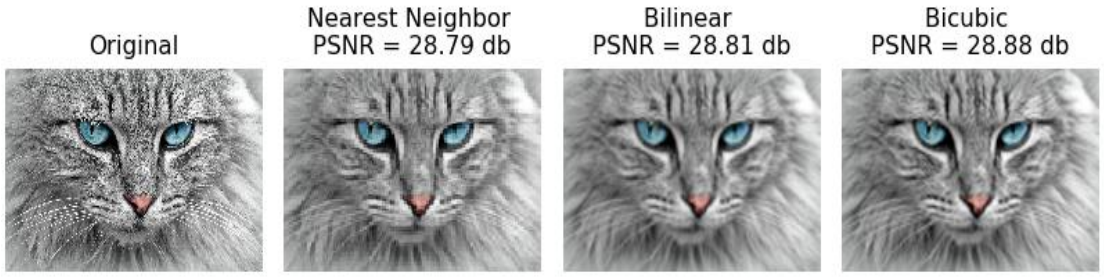
$SSIM(x,y) = [l(x,y)^\alpha * c(x,y)^\beta * s(x,y)^\gamma]$, $SSIM = 1$ αν οι εικόνες x, y είναι όμοιες

Μέσα από την μηχανική μάθηση, εξετάζεται αν είναι εφικτό να επιτευχθεί αποθορυβοποίηση (denoising), όπως και κατά πόσο περισσότερο θα βελτιώσει τα αποτελέσματα σε σχέση με τις τεχνικές επεξεργασίας εικόνας που αναφέρθηκαν. Η αποθορυβοποίηση εικόνων με συνελικτικά δίκτυα, παράλληλα με την υπερ-ανάλυση εικόνων (super resolution) όπου για να αυξήσεις τις διαστάσεις μια εικόνας, ουσιαστικά προσθέτεις θορυβώδη pixel, έχει απασχολήσει αρκετά την επιστημονική κοινότητα που ασχολείται σε αυτόν τον κλάδο.

Το κλασσικό παράδειγμα που αναφέρεται συνήθως (κυρίως για την υπερ-ανάλυση εικόνων) για να εξηγήσει την έννοια αυτή, είναι σε αστυνομικές σειρές (CSI κ.α.), η μεγέθυνση και η αύξηση της ποιότητας σε πλάνα από κλειστό κύκλωμα παρακολούθησης, όπου στα ανεπεξέργαστα πλάνα, τα χαρακτηριστικά του εγκληματία εμφανίζονται θολά και δεν βοηθάνε στην ανίχνευσή του. Η υπερ-ανάλυση αυτή του παραδείγματος, μπορεί να γίνει χωρίς την χρήση νευρωνικών δικτύων, με τις τεχνικές που βασίζονται σε ομοιότητες με τα διπλανά pixel (Εικόνα 1-10) όπως για παράδειγμα:

- Κοντινά γειτονικά pixel (Nearest Neighbor), όπου τα κενά pixel που δημιουργούνται έχουν τιμές των διπλανών pixel
- Διγραμμικη (Bilinear), όπου τα pixel είναι η μέση τιμή των διπλανών pixel
- και δικυβική παρεμβολή (bicubic Interpolation), όπου αναλύει 16 γειτονικά pixel (4x4) σε σχέση με την διγραμμική (2x2) και δίνει τα καλύτερα αποτελέσματα οξύτητας σε σχέση με τις προηγούμενες τεχνικές.

Στην παρακάτω εικόνα συγκρίνονται τα αποτελέσματα για τις τεχνικές που αναφέρθηκαν, κάνοντας την αρχική εικόνα σμίκρυνση στο 10% της αρχικής ανάλυσης και μετά πάλι μεγέθυνση στο αρχικό μέγεθος.



Εικόνα 1-10 Η σύγκριση των αποτελεσμάτων για κάθε τεχνική που αναφέρθηκε. Η τεχνική bicubic αποδεικνύει μέσω του μεγαλύτερου PSNR ότι προσφέρει καλύτερα αποτελέσματα. Πηγή αρχικής εικόνας pixabay.com

Βάση της επιστήμης όμως της πληροφορίας, δεν μπορείς να προσθέσεις πληροφορίες που δεν βρίσκονται ήδη στην φωτογραφία (Data processing Inequality)[9]. Μία τέτοια μέθοδος προσθέτει στην εικόνα θολότητα. Δηλαδή το χρώμα, ή οι γραμμές και οι γωνίες που σχηματίζουν τα αντικείμενα που απεικονίζονται, δεν απεικονίζουν το πραγματικό αντικείμενο και προσθέτουν οπτικό θόρυβο στην εικόνα, από τα Pixel που προστέθηκαν (Εικόνα 1-10).

Με την χρήση της εποπτευόμενης μάθησης, μπορούμε να προσθέσουμε πληροφορίες που βρίσκονται ήδη στις φωτογραφίες υψηλότερης ποιότητας και μέσω της εκπαίδευσης να

βελτιώσουν τα αποτελέσματα των παραπάνω τεχνικών. Στα CNN μέσω της εκπαίδευσης, οι εικόνες χαμηλότερης ποιότητας της εισόδου, συγκρίνονται με τις εικόνες υψηλής ποιότητας της εξόδου (Ground Truth), και στην οπισθοδιάδοση (back propagation), διαμορφώνουν τα βάρη ανάλογα με την συνάρτηση κόστους (Loss function) και τη διαφορά ανάμεσα σε είσοδο (input) και έξοδο (Ground Truth) (αλγόριθμος Gradient Descent).

Μέσα από αυτούς τους υπολογισμούς τα CNN διαμορφώνουν τα βάρη, που αναγνωρίζουν τα κατάλληλα πρότυπα για τη βελτίωση ποιότητας (quality enhancement) των φωτογραφιών εισόδου, που μπορεί να είναι απλά πρότυπα όπως αναγνώριση γραμμών, ή γωνιών και τα χρησιμοποιούν για πιο σύνθετα πρότυπα. Σε νέες εικόνες, διαφορετικές από εκείνες του dataset εκπαίδευσης, το εκπαιδευμένο πλέον CNN θα εφαρμόσει αυτά τα βάρη, που στην ουσία είναι κατάλληλοι πίνακες όπως παρουσιάστηκαν και στα φίλτρα, για να αυξήσει την ποιότητα των εικόνων, που πλέον δεν βασίζεται σε τεχνικές προκαθορισμένων φίλτρων ή αλγόριθμων, αλλά σε μαθηματικές τεχνικές που δοκιμάστηκαν επιτυχώς σε ένα μεγάλο dataset με πολύ μικρό σφάλμα και αντιστοίχησαν (mapping) εικόνες χαμηλής ανάλυσης, με μεγαλύτερης ανάλυσης ή καλύτερης ποιότητας.

Αντικείμενο της εργασίας

Ο στόχος της παρούσας εργασίας είναι η βελτίωση ποιότητας των δεδομένων των αισθητήρων (στην προκειμένη περίπτωση αισθητήρων μέτρησης βάθους), προτού αυτά τα δεδομένα χρησιμοποιηθούν από το πρόγραμμα για εκτέλεση εντολής. Στην εργασία αυτή εξετάστηκε η αποθορυβοποίηση σε εικόνες βάθους, από δύο διαφορετικούς αισθητήρες, με διαφορετική τεχνική απεικόνισης και ποιότητα, που απεικονίζουν το ίδιο αντικείμενο. Βασίστηκε σε μεθοδολογίες εποπτευόμενης μάθησης που εξετάστηκαν σε παρόμοια προβλήματα από προηγούμενες έρευνες, ώστε τα αποτελέσματα της εργασίας να είναι αποδοτικότερα σε σχέση με μεθόδους επεξεργασίας εικόνων, σαν αυτές που παρουσιάστηκαν παραπάνω.

Η αποθορυβοποίηση έγινε σε εικόνες βάθους που προήλθαν από τα τρισδιάστατα αρχεία νέφη σημείων, από μια RealSense D345 κάμερα και με νέφη σημείων από τον αισθητήρα ConoPoint-10, που λήφθηκαν με την μέθοδο που αναφέρθηκε σε προηγούμενο κεφάλαιο.

Η επιτυχής βελτίωση ποιότητας σε αισθητήρες χαμηλότερου κόστους, δύναται να χρησιμοποιηθεί σε πολλές εφαρμογές, όπου η ανάγκη για πιο ακριβή δεδομένα δεν μπορεί να υλοποιηθεί με την χρήση καλύτερων (από άποψη υλισμικού) αισθητήρων. Για παράδειγμα ένας ρομποτικός βραχίονας για να σηκώσει ένα αντικείμενο σε απόσταση Z, αν χρησιμοποιεί έναν αισθητήρα βάθους για το ολόγραμμα του αντικειμένου και τις διαστάσεις του, το σφάλμα ανάμεσα στις πραγματικές διαστάσεις και τις μετρούμενες από τον αισθητήρα, πρέπει να είναι ελάχιστο για την σωστή εκτέλεση. Ένας φθηνότερος αισθητήρας με μεγαλύτερο σφάλμα στη μέτρηση και βελτίωση ποιότητας στην έξοδο του, μπορεί σε εφαρμογές όπως του παραπάνω παραδείγματος, να έχει πλεονεκτήματα στη χρήση του, αν η θέση του αισθητήρα, ή η διεργασία του μηχανήματος προκαλεί φθορά στους αισθητήρες, με αποτέλεσμα την υψηλού κόστους, συχνή αντικατάσταση τους.

Έτσι μέσα από την εργασία ερευνάται, τα αποτελέσματα που θα είχε η βελτίωση ποιότητας και κατά πόσο μπορούν να αντικαταστήσουν, ή να υποβοηθήσουν κάμερες που περιέχουν θόρυβο και ελλιπή δεδομένα, όπως οι stereo depth RealSense κάμερες. Οι κάμερες αυτές

είναι ευκολότερα προσιτές στο καταναλωτικό κοινό και μπορούν να χρησιμοποιηθούν σε εφαρμογές που χρησιμοποιούν άμεσα δεδομένα βάθους, ή έμμεσα για βέλτιστη λειτουργία και απόδοση. Εφαρμογές όπως: αυτοματισμοί θέρμανσης, ψύξης κτλ., μπορούν να βελτιώσουν την απόδοση τους χρησιμοποιώντας δεδομένα βάθους. Η χρήση ακριβών laser όπως πιθανόν θα ανέβαζε πολύ το κόστος, ενώ επιπλέον το ConoPoint-10 θα αυξήσει πολύ και τον απαιτούμενο χρόνο σκαναρίσματος (μέσος χρόνος 20 λεπτά) και ίσως να δημιουργηθεί μια δυσαναλογία απόδοσης-κόστους. Μέσα από φθηνότερους αισθητήρες όπως της RealSense που η έξοδος τους θα βελτιώνεται από λογισμικό βαθιάς μάθησης, θα μπορούν να έχουν θεωρητικά απλούστερο υλισμικό (hardware).

Προηγούμενες έρευνες

Στον τομέα της μηχανικής όρασης, η αντιστοίχιση εικόνων χαμηλότερης ανάλυσης ή (ποιότητας) με υψηλότερης ανάλυσης είναι ένα κλασσικό πρόβλημα. Στην δημοσιευμένη έρευνα «Image Super-Resolution Using Deep Convolutional Networks» [3] οι συγγραφείς πρότειναν μια αρχιτεκτονική μοντέλου για την υπερ-ανάλυση που ονομάστηκε SRCNN, που είχε καλύτερα αποτελέσματα από την Bicubic τεχνική. Στην έρευνα χρησιμοποίησαν ένα συνελικτικό δίκτυο τριών στρωμάτων, με μέγεθος φίλτρων 9-1-5 και αριθμό φίλτρων 64-32-1. Ως δεδομένα εισόδου χρησιμοποίησαν εικόνες, που αρχικά μεγεθύνθηκαν (upscale) με την τεχνική bicubic και μετά σμικρύνθηκαν (downscale) στο αρχικό μέγεθος, ώστε να έχουν θόρυβο και θολότητα (blur). Για δεδομένα εξόδου (Ground Truth) χρησιμοποιήθηκαν οι αρχικές εικόνες.

Η μεθοδολογία τους βασίστηκε σε τρείς λειτουργίες για την αντιστοίχιση χαμηλής ανάλυσης εικόνων, με την υψηλής:

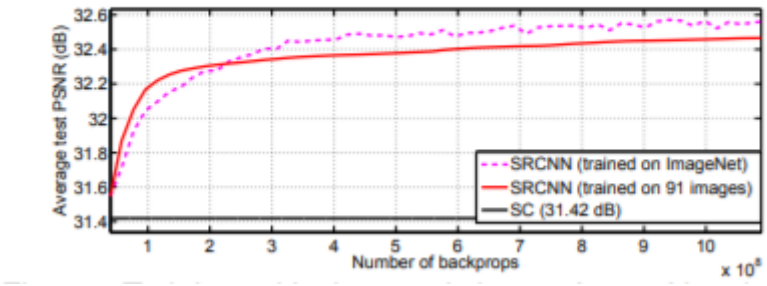
- Εξαγωγή προτύπων (Patch extraction and representation): όπου αποσπάει πρότυπα μέσω των φίλτρων του CNN από την εικόνα χαμηλής ανάλυσης. Τα πρότυπα αυτά παρουσιάζονται σαν υψηλών διαστάσεων διάνυσμα (vector).
  Αυτό η λειτουργία μαθηματικά εκφράζεται ως:

$$fn(Y) = max(0, w_n * Y + B_n)$$

  Όπου:

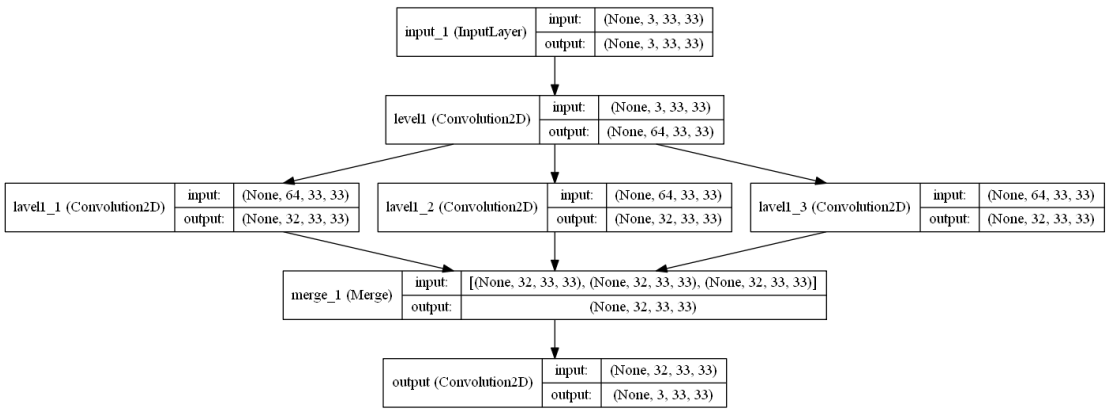
  fn = κρυμμένο στρώμα (layer)
  Y η εικόνα low resolution ή τα διανύσματα για επόμενα στρώματα
  w_n, B_n τα φίλτρα και τα βάρη
  * convolution
  w_n αντιστοιχεί σε:
    - n_n αριθμός των φίλτρων με διαστάσεις $c \times f_n \times f_n$ όπου c τα channels των εισόδων και f ύψος, πλάτος.
- Μη γραμμική αντιστοίχιση (Non-linear mapping): όπου μη γραμμικά αντιστοιχούνται τα διανύσματα (vectors) υψηλών διαστάσεων σε άλλα υψηλών διαστάσεων διανύσματα ώστε να περιληφθούν και άλλα πρότυπα.
  Το w_n πλέον είναι διαστάσεων, $n_{n-1} \times f_n \times f_n$ και B_nείναι B_{n-1}διαστάσεων
- Ανακατασκευή της εικόνας μέσα από τα πρότυπα ώστε να αντιστοιχεί στην εικόνα υψηλής ανάλυσης

Στο SRCNN χρησιμοποιήθηκαν δύο πακέτα δεδομένων (dataset) για την εκπαίδευση του δικτύου. Το ένα σχετικά μικρό με 91 εικόνες και ένα μεγάλο με 395,909 εικόνες από το ILSVRC 2013 ImageNet detection training dataset [11]. Από αυτά τα dataset χρησιμοποίησαν υπο-εικόνες μεγέθους 33 pixel. Βάση αυτού οι 91 εικόνες κατέληξαν 24,800 εικόνες με stride 14. Ενώ στο ImageNet με stride 33, δημιουργήθηκαν περισσότερες από 5,000,000 εικόνες. Για την εκπαίδευση χρησιμοποιήθηκε η συνάρτηση κόστους MSE (Mean Square Error) και αλγόριθμος βελτιστοποίησης SGD (stochastic gradient descent). Για έλεγχο επίδοσης (metrics) χρησιμοποιήθηκε η συνάρτηση PSNR (Peek Signal to Noise Ratio), όπως επίσης η συνάρτηση SSIM. Τα αποτελέσματα της εκπαίδευσης φαίνονται στην παρακάτω εικόνα.



Εικόνα 1-11 Η εκπαίδευση των δύο dataset το SRCNN [3]

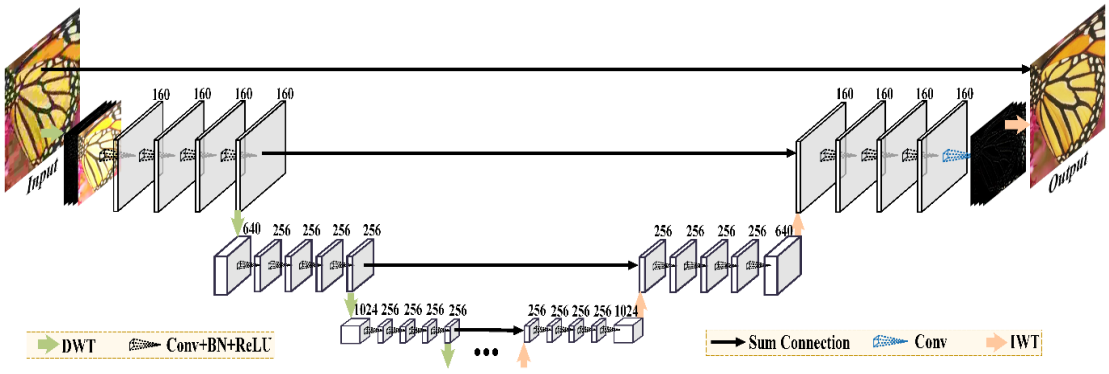
Έχουν ερευνηθεί και άλλα δίκτυα επιβλεπόμενης μάθησης όπως το ESRCNN (Expanded Super Resolution CNN) [29] που φαίνεται στην (Εικόνα 1-12). Προσπαθούν να αναπτύξουν βαθύτερα δίκτυα, με κόμβους στρωμάτων με βάση το SRCNN, αλλά δεν καταφέραν να έχουν καλύτερη απόδοση από αυτό. Άλλωστε στα πειράματα των ερευνητών του SRCNN δοκίμασαν μοντέλα με περισσότερο βάθος που αστόχησαν σε σχέση με την αρχιτεκτονική (9-1-5).



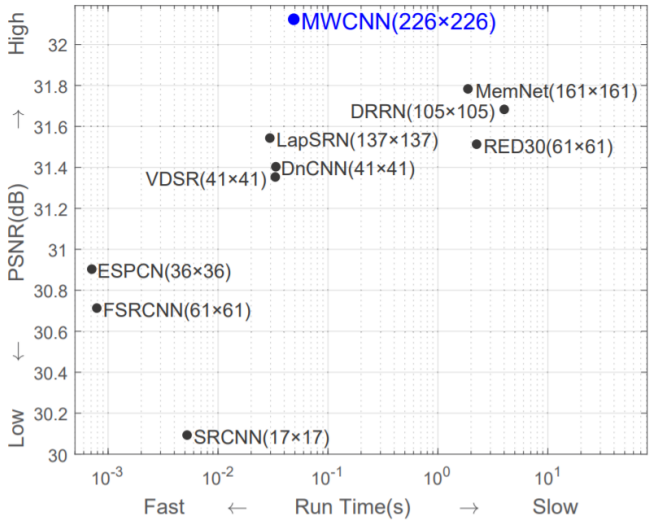
Εικόνα 1-12 ESRCNN [29] Διαφορετική αρχιτεκτονική βασισμένη στο SRCNN που προσπάθησε να βελτιώσει τα αποτελέσματα του

Στην έρευνα [7] δοκίμασαν ένα δίκτυο CNN με την τεχνική Multi-level Wavelet-CNN για αποκατάσταση εικόνας (Image restoration) που χρησιμοποιεί τεχνική discrete wavelet transform (DWT) για να αντικαταστήσει την τεχνική pooling, όπου σε αντίθεση με την

τεχνική pooling, είναι αναστρέψιμη κατά την υπο-δειγματοληψία της εικόνας εισόδου, με την inverse wavelet transform (IWT) ως μαθηματική συνάρτηση κατά την υπερδειγματοληψία των δεδομένων της αρχικής εικόνας στην εικόνα ground truth. Το δίκτυο αυτό (Εικόνα 1-13) μπορεί να επεξηγηθεί και ως μια γενίκευση του Dilated Filtering και της δειγματοληψίας και μπορεί να χρησιμοποιηθεί σε πολλές εφαρμογές αποκατάστασης εικόνας. Όπως φαίνεται και στην Εικόνα 1-14, πέτυχε αρκετά υψηλό PSNR σε σύγκριση με μοντέλα άλλων paper για αποκατάσταση εικόνων, συνδυάζοντας ταχύτερη εκπαίδευση.



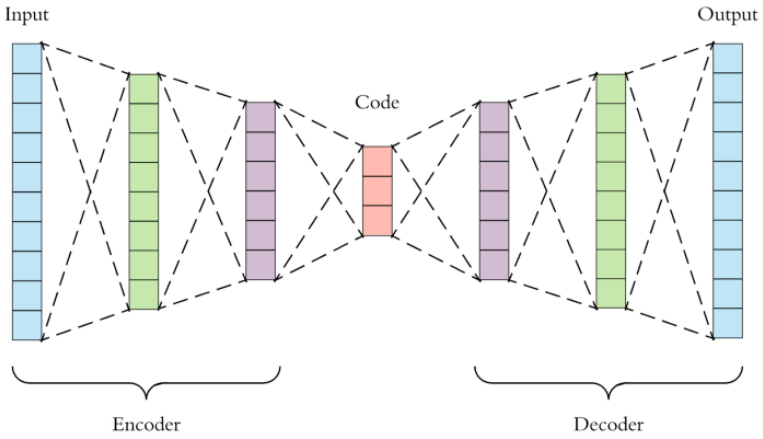
Εικόνα 1-13 Η αρχιτεκτονική του Multi-level Wavelet-CNN [7]



Εικόνα 1-14 Το PSNR του δικτύου MWCNN σε σύγκριση με άλλα δίκτυα με βάση το PSNR και το υπολογιστικό κόστος. [7]

Τεχνικές όπως η παραπάνω, που κάνουν υποδειγματοληψία (down sample) και υπερδειγματοληψία (up sample) κατάφεραν αρκετά ικανοποιητικά αποτελέσματα σε παρόμοια προβλήματα. Μεγαλύτερη απόδοση (state-of-the-art) φαίνεται τα έχουν τεχνικές μάθησης χωρίς επίβλεψη, ή με ημι-επίβλεψη (semi-supervised) με την χρήση SRGANs (Super Resolution Generative Adversarial Networks)[13] και στα Autoencoders (αυτό-κωδικοποιητές)[14][16] όπου είναι δίκτυα που προσπαθούν να αντιγράψουν την είσοδο στην έξοδο.

Οι τεχνικές SRGANs δεν δοκιμάστηκαν γι’ αυτό και δεν αναφέρονται στην εργασία, αντίθετα έγινε προσπάθεια υλοποίησης αλγόριθμων autoencoder για την σύγκριση των αποτελεσμάτων. Μοντέλα autoencoder φαινομενικά χωρίζονται σε δύο δίκτυα, τον κωδικοποιητή (encoder) για σμίκρυνση των διαστάσεων και τον αποκρυπτογράφο (decoder) για μεγέθυνση των διαστάσεων. Η μείωση των διαστάσεων, μπορεί να πραγματοποιηθεί μέσω των συνελικτικών υπολογισμών, χωρίς προσθήκη pixel (padding), η με εξαγωγή των μέγιστων (επίσης μέσων, ή ελάχιστων) τιμών, με την τεχνική (pooling). Για την μεγέθυνση των διαστάσεων, είτε μπορεί να γίνει μεγέθυνση με εισαγωγή pixel, είτε με αντίστροφους συνελικτικούς υπολογισμούς (transpose convolution). Κατά την μεγέθυνση (decoder) το δίκτυο προσπαθεί να βρει τα κατάλληλα βάρη, για να ανακατασκευάσει και να αντιστοιχίσει την εικόνα εισόδου, στην εικόνα εξόδου. Χαρακτηριστικό των autoencoders είναι το βάθος που ονομάζεται και “bottle neck”, γιατί θυμίζει το στόμιο ενός μπουκαλιού και δηλώνει τον βαθμό μεγέθυνσης και σμίκρυνσης. Στην παρακάτω εικόνα αναπαρίσταται το σχήμα ενός autoencoder



Εικόνα 1-15 Σχήμα ενός autoencoder. [40]

2. Αποθορυβοποίηση εικόνων Βάθους

Η μεθοδολογία που εξετάστηκε βασίστηκε κυρίως για πειραματισμό στο μοντέλο SRCNN [3] που αναφέρθηκε στο κεφάλαιο 0 και πέτυχε αξιόλογα αποτελέσματα σε παρόμοιο πρόβλημα. Εξετάστηκαν οι αρχιτεκτονικές που παρουσίασαν οι συγγραφείς στην ερευνά τους, όπως επίσης και μερικές διαφορετικές, πιο βαθιές αρχιτεκτονικές. Διαφέρει από τη μεθοδολογία του SRCNN ως προς τα δεδομένα, καθώς η χρήση δεδομένων από διαφορετικούς αισθητήρες (των Laser και της κάμερας RGBD) διαφέρουν πολύ ως προς τον θόρυβο και την ποιότητα σε σχέση με το dataset του SRCNN.

Στη διπλωματική χρησιμοποιήθηκε ένα πακέτο δεδομένων (dataset) από 95 αντικείμενα κοινής χρήσης, που σαρώθηκαν από τη διάταξη σάρωσης με τον αισθητήρα Conopoint-10 που αναφέρθηκε και την κάμερα RealSense D435. Και συγκεκριμένα:

- 95 εικόνες βάθους υψηλής ποιότητας (Conopoint) και 95 χαμηλής ποιότητας (RealSense), διαστάσεων 63*96 Pixels. (Στην εργασία χρησιμοποιήθηκαν και οι όροι HR/LR για αναφορά σε υψηλής και χαμηλής ποιότητας αντίστοιχα από τον αγγλικό όρο High and Low Resolution)
- Δοκιμάστηκαν τεχνικές επαύξησης δεδομένων, για να αυξηθεί ο αριθμός των εικόνων και διάφορες αρχιτεκτονικές δικτύου. Η πορεία περιγράφεται στα επόμενα κεφάλαια.

Η πορεία και η μεθοδολογία επεξηγείται στο κεφάλαιο των αποτελεσμάτων.

2.1.Προ-επεξεργασία δεδομένων

Στην εποπτευόμενη μάθηση ένα πολύ σημαντικό κομμάτι είναι η προ-επεξεργασία των δεδομένων. Τα δεδομένα που θα δοθούν στους αλγόριθμους αν εμφανίζουν κενά, ή λάθος μετρήσεις θα προκαλέσουν σφάλματα και αποκλίσεις στα αποτελέσματα των αλγορίθμων.

Στο κεφάλαιο αυτό παρουσιάζονται η μεθοδολογία για την επεξεργασία των ανεπεξέργαστων (raw) αρχείων προτού εφαρμοστούν στο CNN σαν εικόνες βάθους.

Το πακέτο δεδομένων αποτελείται από αρχεία νέφη σημείων (.ply), δηλαδή αρχεία όπου περιέχουν τα σημεία xyz όπως φαίνεται στην Εικόνα 2-1. Τα νέφη σημείων (point clouds) χωρίζονται σε αυτά της RealSense και του ConoPoint-10 , που θα χρησιμοποιηθούν για το τελικό πακέτο δεδομένων, ως εικόνες βάθους στο τεχνητό νευρωνικό δίκτυο.

```
ConoPoint-10 pointcloud με σχήμα συστοιχίας (9476, 3)
Αποτελείται από 9476 xyz τιμές,

[[   0.       0.    -172.901]
[   0.       0.5   -159.59 ]
[   0.       1.    -159.64 ]
...
[  49.5      1.    -174.186]
[  49.5      0.5   -174.205]
[  49.5      0.    -174.213]]

RealSense pointcloud με σχήμα συστοιχίας (407040, 3)
Αποτελείται από 407040 xyz τιμές
```

Εικόνα 2-1 Παράδειγμα από τα αρχεία νέφη σημείων (point clouds) με κατάληξη (.ply) για ένα ίδιο αντικείμενο, της Real Sense και του Laser Scanner

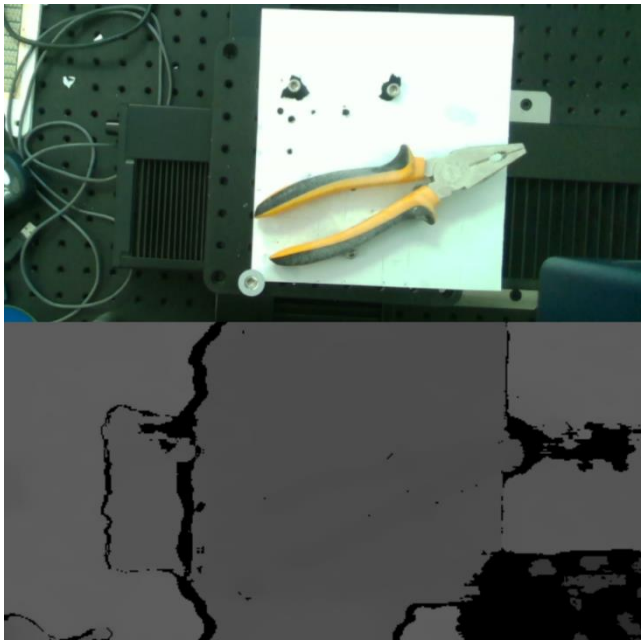
Τα νέφη σημείων της Εικόνα 2-1 έχουν:

- Νέφος σημείων laser με σύνολο 9476 points με τιμές x,y,z, 0-49.5 με βήμα 0.5 και μέτρηση απόστασης z
- Το νέφος σημείων της RealSense με 407040 Points x,y,z

Τα ανεπεξέργαστα αρχεία νέφους σημείων με τα σημεία xyz της εικόνας 2-1, μπορούμε να τα ανοίξουμε και να τα δούμε σαν τρισδιάστατα αρχεία με την βιβλιοθήκη Open3d, όπως φαίνεται και στην Εικόνα 2-2.

Εικόνα 2-2 Τα νέφη σημείων για δύο ίδια αντικείμενα. Με πράσινο χρώμα της RealSense και με πορτοκαλί του Laser όπως φαίνεται με την μέθοδο visualization.draw_geometries από την βιβλιοθήκη της Open3d.

Τα ανεπεξέργαστα αρχεία της RealSense περιλαμβάνουν την rgb εικόνα και την εικόνα βάθους (Greyscale). Από αυτά τα αρχεία μπορούμε να δημιουργήσουμε νέφη σημείων (για να τα συγκρίνουμε με αυτά από το Conopoint-10), μέσω δικού της λογισμικού που παρέχεται από τον κατασκευαστή. Τα αρχεία φαίνονται στην Εικόνα 2-3, με την εικόνα βάθους να έχει επεξεργαστεί μέσω προγράμματος επεξεργασίας εικόνων για να είναι διακριτή. Η επεξεργασία έγινε στα επίπεδα φωτεινότητας και αντίθεσης.

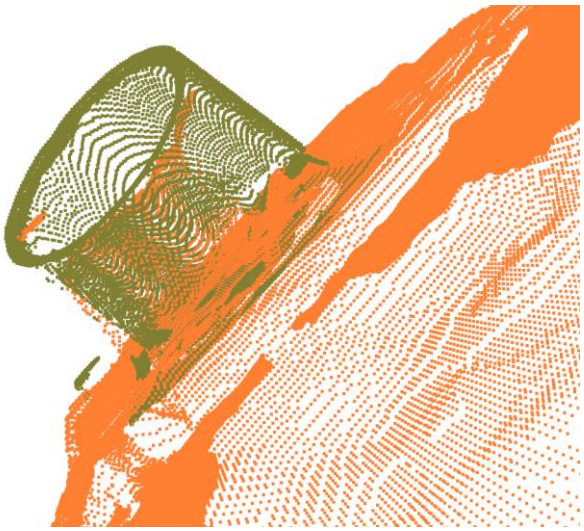


Εικόνα 2-3 Πάνω η εικόνα RGB της RealSense που απεικονίζει μια πένσα και κάτω η εικόνα βάθους (Greyscale) για το ίδιο αντικείμενο.

Πριν μετατραπούν τα νέφη σημείων σε εικόνες βάθους, έγινε επεξεργασία σε:

1. Ευθυγράμμιση στους άξονες x, y, z
2. Φίλτρο στον άξονα z του laser
3. Περικοπή του νέφους σημείων της RealSense στα όρια xyz του νέφους σημείων του laser.

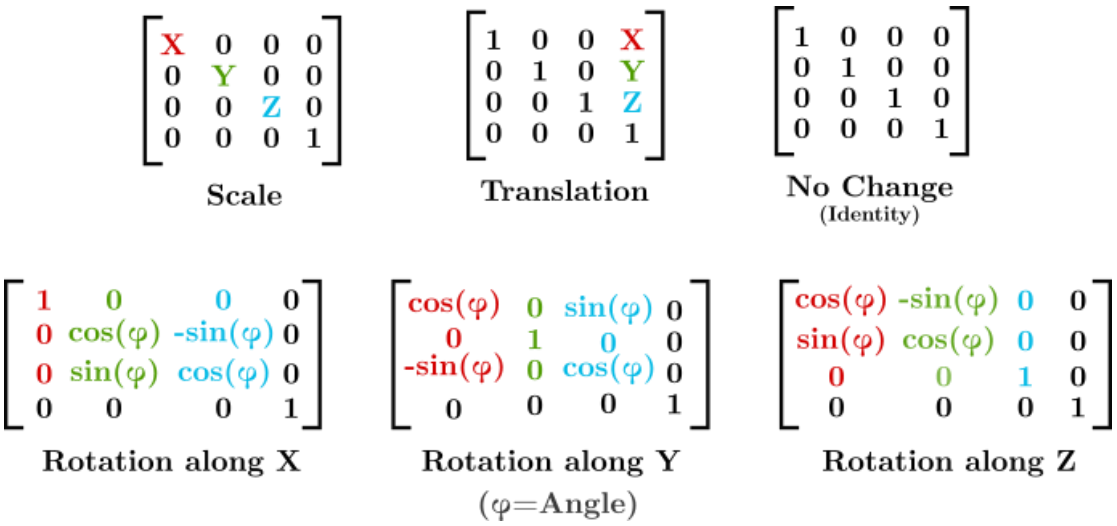
Ευθυγράμμιση στους άξονες xyz



Εικόνα 2-4 Η ευθυγράμμιση των δύο νέφη σημείων (πορτοκαλί της RealSense και πράσινο του Conopoint)

Έχοντας δύο νέφη σημείων με διαφορετικές συντεταγμένες xyz (όπως φαίνεται και στην Εικόνα 2-1), πρέπει να υπολογιστεί ο αλγεβρικός πίνακας T (matrix), που θα περιλαμβάνει τις μεταβλητές περιστροφής (rotation) και μεταφοράς (translation) , για να ευθυγραμμιστεί το ένα από τα δύο νέφη σημείων, με σημείο αναφοράς το σύστημα συντεταγμένων του

άλλου (άκαμπτη εγγραφή ή Rigid registration). Ο πίνακας T συνήθως είναι 4*4 και χρησιμεύει για rotation (περιστροφή), translation (μετακίνηση) και scaling (κλιμάκωση) πολλαπλασιάζοντας βάση της θεωρίας των πινάκων τις σημεία του νέφους με τον πίνακα T. Οι μεταβλητές για κάθε ενέργεια φαίνονται στην παρακάτω εικόνα:



Εικόνα 2-5 Πίνακας 4*4 και οι μεταβλητές για περιστροφή (rotation) στους 3 άξονες, μεταφορά (translation) και κλιμάκωση (scale) [36]

Το 4*4 matrix μπορεί να βρεθεί και με την βοήθεια του ανοιχτού λογισμικού MeshLab που χρησιμοποιείται για προβολή και επεξεργασία τρισδιάστατων αρχείων. Φορτώνοντας τα δύο αρχεία, στο μενού align γίνεται επιλογή 6 σημείων από κάθε νέφος για αντιστοίχιση ανάμεσα τους. Η μέθοδος αυτή ονομάζεται point set registration[37] και μεταφράζεται ως «εγγραφή σετ σημείων». Το πρόβλημα αυτό συνοψίζεται ως: η μεταλλαγή προσανατολισμού και θέσης, ενός εκ των δύο πεπερασμένων σετ σημείων $\{C, R\}$ ως προς το άλλο, μέσω ενός αλγεβρικού πίνακα T που θα ελαχιστοποιεί την απόσταση (ευκλείδεια απόσταση) ανάμεσα στα σημεία και θα προσφέρει την καλύτερη ταύτιση θέσεων ανάμεσά τους. Ο μαθηματικός τύπος που εκφράζει το παραπάνω πρόβλημα είναι:

$$T^* = \arg\min_{T \in \tau} dist(T(C), R)$$

Όπου:

T^* : Ο ιδανικός αλγεβρικός πίνακας ώστε το νέφος σημείων C να ευθυγραμμιστεί ιδανικά με το R.
τ : Όλοι οι πιθανοί πίνακες μετασχηματισμού που θα εξεταστούν από τον αλγόριθμο βελτιστοποίησης, για να βρεθεί ο ιδανικός αλγεβρικός πίνακας.
$T(C)$: Ο πίνακας που θα εφαρμοστεί στο νέφος σημείων C για αντιστοίχιση στο νέφος R.
$arg\ min$: Οι τιμές για τα ελάχιστα της συνάρτησης (argument of the minimum)

Το Meshlab υπολογίζει τον πίνακα T και αποθηκεύεται σε αρχείο κειμένου. Υπάρχουν αρκετοί αλγόριθμοι που χρησιμοποιούνται για τον υπολογισμό του T^*, όπως ο αλγόριθμος Iterative closest Point και περιγράφονται στην σχετική σελίδα της Wikipedia [37]. Στην

Εικόνα 2-2 φαίνονται τα δύο νέφη χωρίς την ευθυγράμμιση, ενώ στην Εικόνα 2-4 φαίνεται το τελικό αποτέλεσμα ευθυγράμμισης.

Ο αντίστροφος πίνακας T^{-1} μπορεί να χρησιμοποιηθεί για την αντίστροφη επεξεργασία ευθυγράμμισης των συντεταγμένων του *R* (RealSense) ως προς το *C* (Conopoint) και επιλέχθηκε και στον κώδικα της εργασίας. Η επιλογή αυτή, της ευθυγράμμισης του RealSense ως προς το σύστημα συντεταγμένων του laser, έγινε λόγω της ευκολίας ανάγνωσης των συντεταγμένων x,y (εικόνα 2-1) του laser.

Για τον πολλαπλασιασμό των σημείων του νέφους σημείων με σχήμα (numpy.shape(407040 * 3)) με τον 4*4 πίνακα, πρέπει βάση της θεωρίας των πινάκων, να έχουν ίδιο μέγεθος στήλης με γραμμή (column to row). Έτσι προστέθηκε μια στήλη από μονάδες ίδιου μεγέθους στο νέφος σημείων (ομογενή σημεία) για τον πολλαπλασιασμό της συστοιχίας (numpy.array) με σχήμα(407040 * 4) ή οποία στήλη αφαιρείται μετά.

Με την ευθυγράμμιση επιτεύχθηκε σε μεγάλο βαθμό η ταύτιση θέσεων στην 2D εικόνα των αντικείμενων hr και lr που είναι αρκετά σημαντικός παράγοντας για τη σωστή αποθορυβοποίηση με επιβλεπόμενη μάθηση.

Φίλτρο στον άξονα z

Πριν την περικοπή του RealSense στα όρια του laser, πρέπει να αφαιρεθεί ο θόρυβος που παρουσιάζει το νέφος σημείων του laser, για πιο ακριβή μετατροπή σε εικόνες βάθους. Η βιβλιοθήκη της open3d δίνει μερικές λύσεις για φιλτράρισμα όπως :

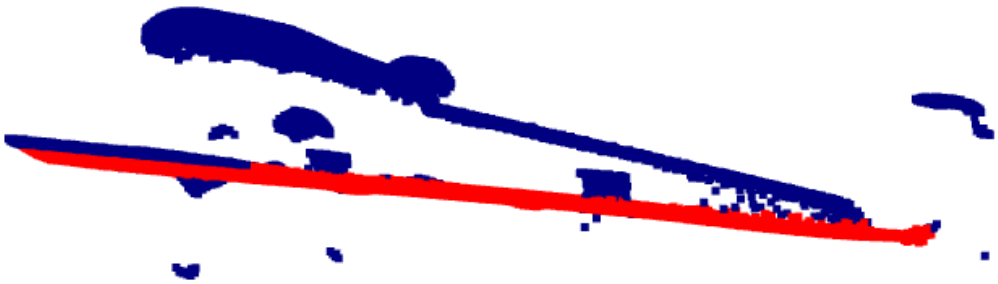
- στατιστική αφαίρεση ακραίων τιμών (statistical_outlier_removal), για την αφαίρεση σημείων που απέχουν μακριά από γειτονικά σημεία.
- Αφαίρεση ακραίων τιμών με ακτίνα (radius_outlier_removal), για την αφαίρεση σημείων με λίγα γειτονικά σημεία σε μια επιλεγόμενη ακτίνα. [28]

Στην συγκεκριμένη εφαρμογή για τα νέφη του Laser, επιλέχθηκε μια διαφορετική προσέγγιση με την βοήθεια της μεθόδου segment_plane της βιβλιοθήκης open3d. Όπως φαίνεται και στις παρακάτω εικόνες, με την μέθοδο αυτή, που χρησιμοποιεί τον αλγόριθμο RANSAC (Random sample consensus) για τον υπολογισμό των σημείων του επιπέδου (inliers), ανιχνεύει τα σημεία επιπέδου και τα σημεία μακριά από το επίπεδο του νέφους σημείων. Όπου η γενική εξίσωση επιπέδου [α,β,γ,δ]

$$\alpha x + \beta y + \gamma z + \delta = 0$$

Και προκύπτει η εξίσωση πεδίου:

$$0.0x + (-0.01y) + 1.0z + (-457.32) = 0$$



Εικόνα 2-6 Ένα νέφος σημείων που απεικονίζει ένα κατσαβίδι. Με μπλε χρώμα παρουσιάζονται τα σημεία μακριά από το επίπεδο (outliers) και με κόκκινο το επίπεδο. Βάση του αλγόριθμου RANSAC

Βάση της τμηματοποίησης αυτής, η μέση τιμή (numpy.average) των σημείων του επιπέδου (σημεία με κόκκινο χρώμα Εικόνα 2-6) χρησιμοποιήθηκε ως όριο, με τιμές μεγαλύτερες (numpy.max) από την μέση (μεγαλύτερες καθώς στον άξονα z η μεγαλύτερη τιμή είναι η πιο μακρινή από τον αισθητήρα) να λαμβάνονται ίσες με την μέση του επιπέδου (Εικόνα 2-6). Μικρότερες τιμές από το επίπεδο που ίσως να παρουσιάζεται θόρυβος δεν φιλτραρίστηκαν (Εικόνα 2-7)

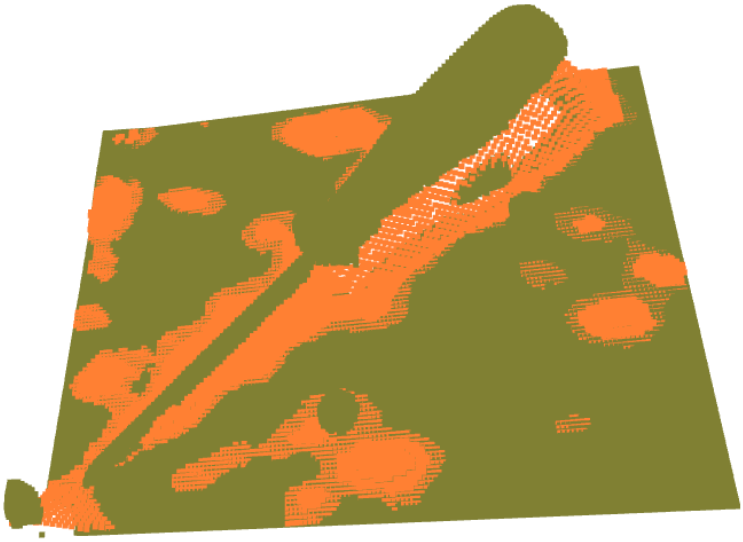
Εικόνα 2-7 Το κατσαβίδι της Εικόνα 2-6 έχοντας φιλτραριστεί για τις μεγαλύτερες τιμές στον άξονα z

Περικοπή του νέφους σημείων της RealSense

Όπως φάνηκε και στην Εικόνα 2-2 το τελευταίο βήμα πριν την μετατροπή σε δισδιάστατες εικόνες βάθους, είναι η περικοπή του νέφους σημείων της RealSense, για να καταλήξει στα όρια xyz του laser.

Για την περικοπή λήφθηκαν τα ελάχιστα και τα μέγιστα όρια xyz του laser, μέσω των μεθόδων της open3d .get_min_bound() και get_max_bound() αντίστοιχα. Και με τη βοήθεια της numpy.logical_and() κρατήθηκαν τα σημεία xyz της RealSense.

Στην Εικόνα 2-8 φαίνεται το νέφος σημείων για το κατσαβίδι που θα χρησιμοποιηθεί ως μέρος του dataset μετά από την προ-επεξεργασία.



Εικόνα 2-8 Παράδειγμα από δύο νέφη σημείων για το ίδιο αντικείμενο, μετά την προ-επεξεργασία. Με πράσινο χρώμα του Laser και πορτοκαλί της RealSense.

Από νέφος σημείων σε δισδιάστατη εικόνα βάθους

Τα νέφη σημείων μετατράπηκαν σε δισδιάστατες (2D) εικόνες βάθους (depth map) με τιμές x,y,z να αντιστοιχούν σε:

- Για τον άξονα z: 0-255 τιμές 'Grey'
- Για τον άξονα x: 96
- Για τον άξονα y: 63

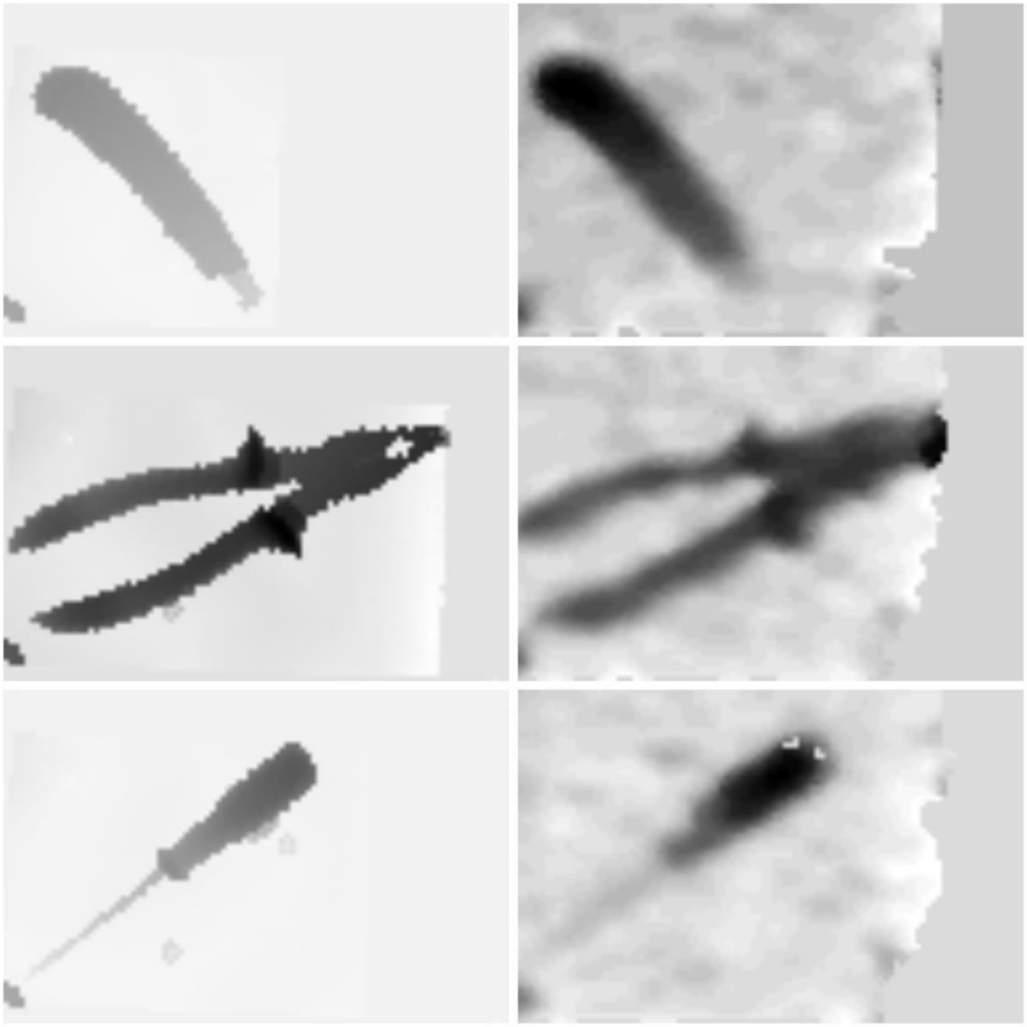
Η αντιστοίχιση έγινε με την γραμμική απεικόνιση (Linear map):

$$d = \frac{(x - in_{min}) * (out_{max} - out_{min})}{(in_{max} - in_{min}) + out_{min}}$$

Όπου:

d: βάθος pixel, ακέραιος (τύπου int) που προκύπτει με την διαίρεση // στην python
x: η τιμή προς αντιστοίχιση
in_{min} : οι ελάχιστες τιμές για τα σημεία xyz των point clouds
in_{max} : οι μέγιστες τιμές για τα σημεία xyz των point clouds
out_{min} : οι minimum τιμές για τις εικόνες βάθους
out_{max} : οι maximum τιμές για τις εικόνες βάθους

Τα 63x96 pixels επιλέχθηκαν για να καλύψουν όλο το σετ δεδομένων που περιλαμβάνει διαφορετικού μεγέθους μετρήσεις στου άξονες x, y των νεφών σημείων του laser ώστε να μην είναι μικρότερες οι διαστάσεις και δημιουργηθούν κενά pixels.



Εικόνα 2-9 Παράδειγμα των εικόνων βάθους υψηλής ποιότητας του Conopoint (αριστερά) και της χαμηλής ποιότητας της RealSense (δεξιά), που χωρίστηκαν ως hr και lr αντίστοιχα για την εκπαίδευση.

Βάση της Εικόνα 2-9 οι εικόνες hr του Conopoint σε σύγκριση με τις lr της RealSense αποδεικνύουν την διαφορά ποιότητας ανάμεσα τους, ως προς τον θόρυβο της μέτρησης. Οι hr εικόνες έχουν αποδώσει καθαρά το περίγραμμα στα αντικείμενα, με ελάχιστο γενικό θόρυβο (γκρι pixel μακριά από το αντικείμενο) κάτι που δεν ισχύει για τις εικόνες lr, στις οποίες θα γίνει προσπάθεια βελτίωσης και αποθορυβοποίησης μέσω CNNs.

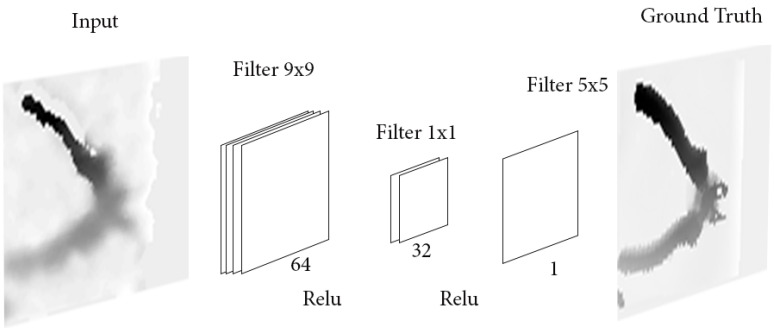
2.2.Το μοντέλο CNN

Οι αρχιτεκτονικές που επιλέχθηκαν για δοκιμές είναι αυτές που πρότειναν και στην έρευνα του SRCNN [3]. Δοκιμάστηκαν και στα 4 dataset που δημιουργήθηκαν. Το βασικό μοντέλο που πρότειναν και αναφέρθηκε και στο κεφάλαιο 0 περιλαμβάνει 3 κρυφά στρώματα με:

- 9-1-5 μέγεθος φίλτρων για τα 3 στρώματα
- 64-32-1 αριθμό φίλτρων για τα 3 στρώματα
- ReLU (Rectified Linear Unit) συνάρτηση ενεργοποίησης για τα πρώτα δύο στρώματα
- Padding = same για να μην συρρικνωθεί το μέγεθος της αρχικής εικόνας.

Το μοντέλο φαίνεται και στην παρακάτω εικόνα (Εικόνα 2-10). Για την ανάπτυξη των αλγορίθμων επιλέχθηκε η API Tensorflow, που περιλαμβάνει την βιβλιοθήκη Keras για την ανάπτυξη αλγορίθμων μηχανικής και βαθιάς μάθησης. Η διαφορά με την υλοποίηση του SRCNN, είναι ότι στο μοντέλο που αναπτύχθηκε σε αυτήν την διπλωματική, χρησιμοποιεί διαφορετικό αλγόριθμο βελτιστοποίησης (Adam optimizer), σε σχέση με τον Stohastic Gradient Descent όπου χρησιμοποίησαν για την ελαχιστοποίηση της συνάρτησης κόστους (Loss Function) στο δίκτυο τους, κάνοντας δοκιμές με διαφορετικό βαθμό εκπαίδευσης (Learning Rate) .

Στο άρθρο [31] ο «Jason Brownlee Phd» επεξηγεί ότι ο «Adam Optimizer (adaptive moment estimation)» είναι ένας αλγόριθμος που προσαρμόζει μόνος του το βαθμό εκπαίδευσης. Ο «Adam» παρουσιάστηκε από τους Diederik Kingma από την OpenAI και Jimmy Ba από το πανεπιστήμιο του Τορόντο [12].



Εικόνα 2-10 Η αρχιτεκτονική του δικτύου που βασίστηκε στο βασικό μοντέλο, που πρότειναν οι συγγραφείς του SRCNN

Τα μοντέλα με τις αρχιτεκτονικές που πρότεινε το paper εκπαιδεύτηκαν με την μέθοδο tensorflow.Keras.Model.fit() με παραμέτρους:

- LR εικόνες σαν έισοδο, με Ground Truth τις HR εικόνες (όπως φαίνεται και στην εικόνα Εικόνα 2-10). Με μέγεθος:
  1 < δέσμη τεμαχίων εικόνων (batch) < σετ δεδομένων εκπαίδευσης
  ώστε να δίνονται είσοδοι περισσότερες από μία εικόνες κατά την εκπαίδευση σε κάθε βήμα για καλύτερα αποτελέσματα στον υπολογισμό κλίσης (Gradient).
- Συνάρτηση κόστους μέσων τετραγώνων (Mean Square Error)

- PSNR σαν μέτρηση επίδοσης (Metric). Για τον έλεγχο της επίδοσης της εκπαίδευσης, με την μέθοδο (tensorflow.image.psnr) και με τις τιμές των Pixels των εικόνων να μετατράπηκαν σε κλίμακα 0 έως 1 από 0-255 (κανονικοποίηση) πριν την είσοδο τους στο δίκτυο.
- Validation Split = 0.05 (διαχωρισμός για σετ επικύρωσης) για το σετ δεομένων 21x32 dataset (b) που πρακτικά σημαίνει ότι ένα 5% των εικόνων της εκπαίδευσης θα χρησιμοποιηθεί σαν σετ επικύρωσης. Στα μικρότερα πακέτα δεδομένων σαν σετ επικύρωσης, επιλέχθηκαν οι εικόνες ελέγχου επίδοσης(test set). Στο τελικό πακέτο δεδομένων επιλέχθηκε validation split 0.2.
- 110 εποχές εκπαίδευσης για τα μικρά dataset , χωρίς σταμάτημα του δικτύου (callback).
- Early stopping (πρόωρο σταμάτημα) σαν Callback, με μέτρηση τον ρυθμό ελαχιστοποίησης της ελάχιστης τιμής του κόστους στο σετ επικύρωσης (validation MSE) και αναμονή (patience) από 10 έως 20 εποχές (epochs). Το πρόωρο σταμάτημα κατά την εκπαίδευση, ανάλογα με την μεταβλητή που θα του ορίσουμε για έλεγχο, ελέγχει για πόσες εποχές δεν υπήρχε βελτίωση στην τιμή της και αν υπερβεί το όριο εποχών που ορίσαμε, τερματίζει την εκπαίδευση και επιστρέφει τα βάρη της καλύτερης εποχής.
- Shuffle = Αληθής (True), για την τυχαία επιλογή εικόνων εκπαίδευσης σε κάθε δέσμη τεμαχίων εικόνων (Batch).

Στο τελευταίο σετ δεδομένων με το μεγαλύτερο πλήθος εικόνων, έγιναν και κάποιες προσπάθειες υλοποίησης autoencoder και άλλων βαθιών αρχιτεκτονικών για πειραματισμό και σύγκριση με τα αποτελέσματα του SRCNN που δοκιμάστηκε εκτενώς στην εργασία.

Αρχιτεκτονικές

Αναλύονται οι αρχιτεκτονικές που πρότειναν στο SRCNN και δοκιμάστηκαν στην εργασία για πειραματισμό. Το βασικό μοντέλο[3] που προτάθηκε, με αριθμό φίλτρων ανά στρώμα (64-32-1) και με μέγεθος φίλτρων (9x9, 1x1, 5x5), είχε αρκετά ικανοποιητικό PSNR συνδυασμό με ταχύτερη εκτέλεση. Καλύτερο PSNR έδωσαν αρχιτεκτονικές με μεγαλύτερο αριθμό φίλτρων αντί του 64-32-1. Όπως επίσης με μεγαλύτερο μέγεθος φίλτρου.

Ωστόσο, βαθύτερα δίκτυα σύμφωνα με τα πειράματα της έρευνας, δεν κατάφεραν ανεβάσουν το PSNR και οι συγγραφείς κατέφθασαν στο συμπέρασμα ότι σε προβλήματα υπέρ-ανάλυσης τα βαθιά (πολλών κρυφών στρωμάτων) CNN δίκτυα δεν είναι αποτελεσματικά, αντίθετα απ’ ότι ισχύει σε προβλήματα κατηγοριοποίησης (classification).

Δοκιμάστηκαν σχεδόν όλες οι αρχιτεκτονικές (Πίνακας 2-1), που εξέτασε στα πειράματά του και το SRCNN και είναι και αυτές που θα παρουσιαστούν για αναλυτικό σχολιασμό. Δεν περιγράφονται αναλυτικά όλα τα πειράματα που έγιναν στην εργασία και θα αναφερθούν στα τελικά συμπεράσματα.

Πίνακας 2-1 Οι αρχιτεκτονικές του SRCNN που επιλέχθηκαν στα dataset για πειραματισμό με το δίκτυο

Αρχιτ/κή	Μέγεθος φίλτρων Layer 1	Μέγεθος φίλτρων Layer 2	Μέγεθος φίλτρων Layer 3	Αριθμός φίλτρων Layer 1	Αριθμός φίλτρων Layer 2	Αριθμός φίλτρων Layer 3

1η	9x9	1x1	5x5	64	32	1
2η	9x9	1x1	5x5	128	64	1
3η	9x9	1x1	5x5	256	128	1
4η	11x11	1x1	7x7	64	32	1
6η	11x11	5x5	7x7	128	64	1

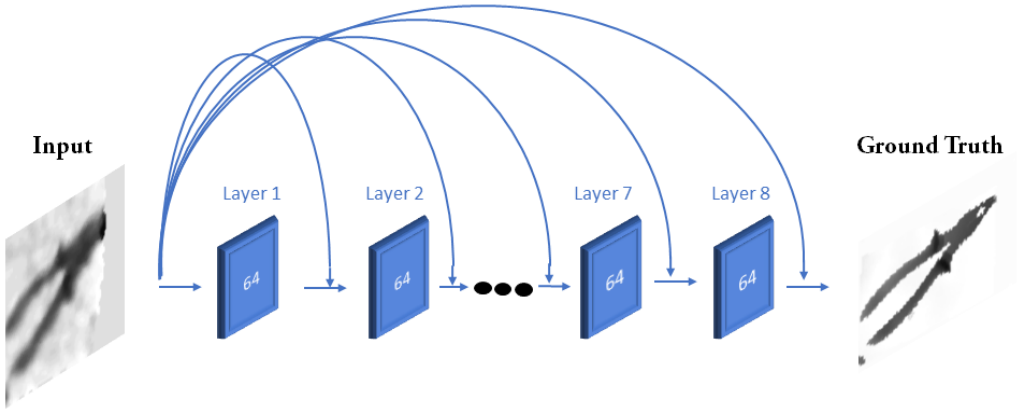
Βαθιές αρχιτεκτονικές

Πέρα από τα πειράματα του SRCNN που έγιναν, δοκιμάστηκαν και πιο βαθιά δίκτυα όσων αφορά την αρχιτεκτονική τους με περισσότερα κρυμμένα στρώματα (layers), παράλειψη συνδέσεων (skip connections), autoencoders κ.α. που βάση της θεωρίας και των αποτελεσμάτων που περιγράφονται και στην έρευνα MWCNN (Εικόνα 1-14) [7], αποδίδουν καλύτερα αποτελέσματα σε τεχνικές βελτίωσης εικόνας (Image restoration). Οι δοκιμές έγιναν στο τελευταίο και μεγαλύτερο dataset που δημιουργήθηκε για πιο αξιόλογα συμπεράσματα. Ο πειραματισμός έγινε με έξοδο (Ground Truth) τις εικόνες hr και είσοδο τις εικόνες lr.

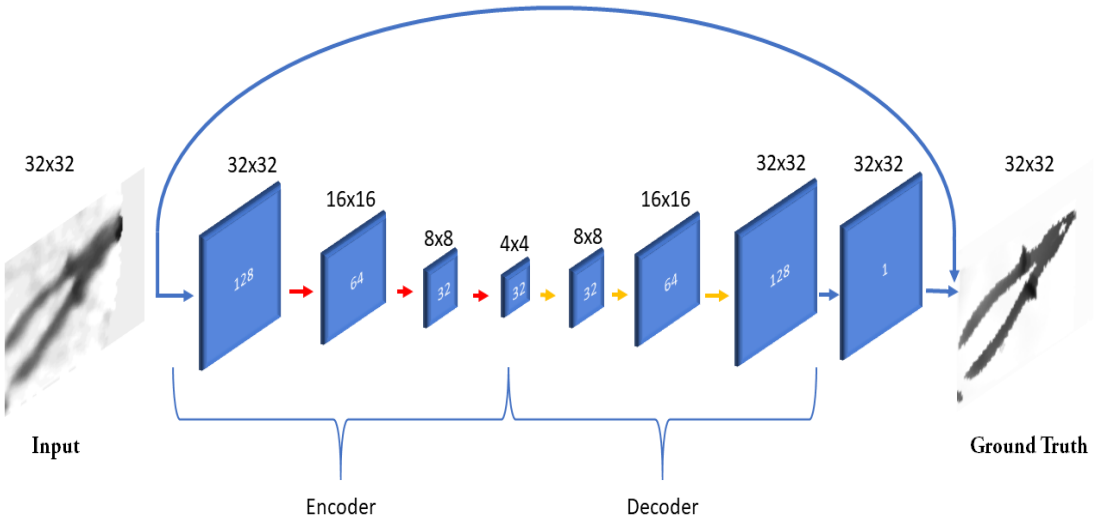
Διαφέρει με την τεχνική MWCNN [7] που επεξηγήθηκε καθώς για την υλοποίηση τους χρησιμοποιήθηκαν layers MaxPooling. Πιο συγκεκριμένα:

- Δημιουργήθηκαν 2 autoencoders
- Χρησιμοποιήθηκαν layers
    - MaxPooling2D 2x2 για την μείωση των διαστάσεων των εισόδων στον encoder
    - Έγιναν δοκιμές με UpSampling2D 2x2 η Conv2DTranspose με stride 2 για την αύξηση των διαστάσεων στον decoder
    - Cropping2D για την περικοπή ύψους της επανακατασκευασμένης εικόνας, καθώς το ύψος των εικόνων σαν περιττός αριθμός στον υποδιπλασιασμό του, μέσα από την keras στρογγυλοποιείται, με αποτέλεσμα στη μεγέθυνση (upscale) να ξεφεύγει από τις αρχικές διαστάσεις.
- Σαν αρχιτεκτονική η υλοποίηση βασίστηκε στο MWCNN για την επιλογή του αριθμού των φίλτρων σε κάθε στρώμα.
- Έγιναν πειράματα με αρχιτεκτονικές όπως αναφέρθηκε, με skip connections (Residual learning) και πυκνών συνδέσεων (dense connections).

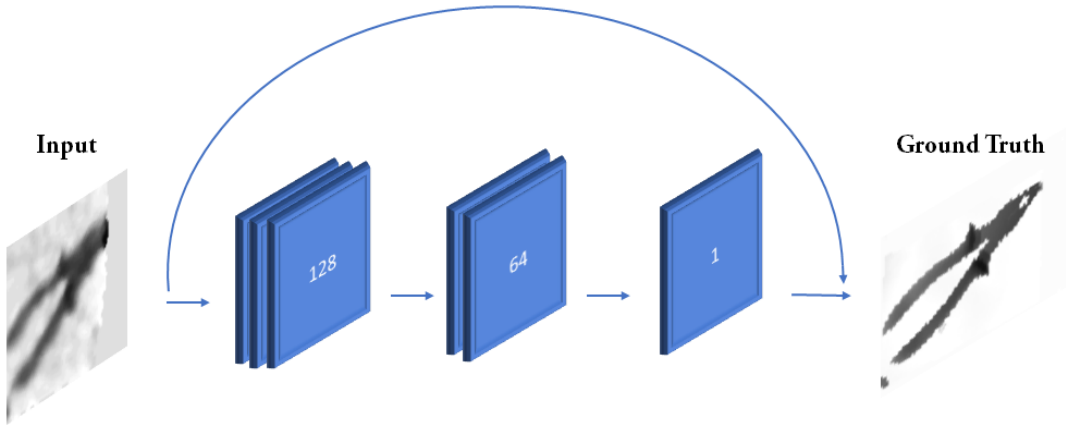
Στις παρακάτω εικόνες, φαίνονται μερικές αρχιτεκτονικές που υλοποιήθηκαν και δοκιμάστηκαν στο τελευταίο σετ δεδομένων και εφαρμόζουν τις μεθόδους με παράλειψη σύνδεσης (skip connection) και πυκνής σύνδεσης (dense connection). Τεχνικές σαν αυτές προσφέρουν ένα «εναλλακτικό μονοπάτι» για τον υπολογισμό κλίσης (gradient) στην οπίσθια διάδοση (backpropagation). Βάση της θεωρίας, σε βαθιές αρχιτεκτονικές ο υπολογισμός κλίσης (gradient) μέσω του κανόνα αλυσίδας (chain rule) αποκτάει τιμή αρκετά χαμηλή, καθώς υπολογίζονται τα βάρη των πρώιμων στρωμάτων. Σε μερικές περιπτώσεις, το gradient μπορεί να γίνει μηδέν με αποτέλεσμα τα πρώιμα στρώματα να μην ανανεώνουν καθόλου τα βάρη τους. Οι συνδέσεις που αναφέρθηκαν, όπου ουσιαστικά προσθέτεις την έξοδο από στρώματα και σε επόμενες συνδέσεις (ή κατευθείαν στην έξοδο), βοηθάνε στο πρόβλημα υπολογισμού gradient αλλά και να μεταβιβαστεί πληροφορία από τα πρώτα στρώματα, στα τελευταία. [42]



Εικόνα 2-11 Δίκτυο 8 στρωμάτων με προσθήκη της εισόδου μετά από κάθε convolution (Dense Connection)



Εικόνα 2-12 Encoder με παράλειψη σύνδεσης (skip connection). Με κόκκινο χρώμα από την πλευρά του encoder συμβολίζονται τα στρώματα max pooling που χρησιμοποιήθηκαν, ενώ με πορτοκαλί στον decoder τα στρώματα Transpose Convolution. Σε κάθε στρώμα φαίνονται και οι διαστάσεις του.



Εικόνα 2-13 SRCNN με παράλειψη σύνδεσης (skip connection).

2.3.Δεδομένα εικόνων του CNN

Από το αρχικό πακέτο δεδομένων (dataset) των 95 εικόνων LR της Real Sense και των 95 εικόνων HR του Laser (Εικόνα 2-9), δοκιμάστηκαν διάφορες τεχνικές επαύξησης δεδομένων που δημιούργησαν διαφορετικού όγκου δεδομένα εικόνων για εκπαίδευση και εξετάστηκαν τα αποτελέσματα τους στο δίκτυο CNN.

Το αρχικό σετ δεδομένων όπως αναφέρθηκε έχει διαστάσεις 63x96 και για να δημιουργηθούν περισσότερες εικόνες, εφαρμόστηκε σαν κύρια μέθοδος επαύξησης δεδομένων (data augmentation) η περικοπή σε μικρότερα κομμάτια των αρχικών εικόνων. Τα βήματα για την περικοπή και η μέθοδος αναφέρεται παρακάτω. Τα αποτελέσματα για κάθε σετ δεδομένων αναλύονται στο κεφάλαιο αποτελέσματα.

Τα βήματα για την δημιουργία των σετ δεδομένων προέκυψαν, κυρίως από τα αποτελέσματα και από τις δοκιμές κάθε προηγούμενων σετ, ώστε να ερευνηθεί σε μεγαλύτερο βαθμό η αποθορυβοποίηση εικόνων βάθους με εποπτευόμενη μάθηση, καθώς και παράγοντες που την επηρεάζουν.

42x64 dataset

Οι 95 εικόνες θεωρητικά είναι ένα πολύ μικρό πακέτο δεδομένων για την ανάπτυξη ενός CNN με ικανοποιητικά αποτελέσματα, έτσι σαν πρώτο βήμα τα δεδομένα χωρίστηκαν σε δεδομένα εκπαίδευσης (training) και ελέγχου (test).

Από τις αρχικές 95 εικόνες, 93 κρατήθηκαν για δεδομένα εκπαίδευσης (training) και 2 για έλεγχο (test) του CNN.

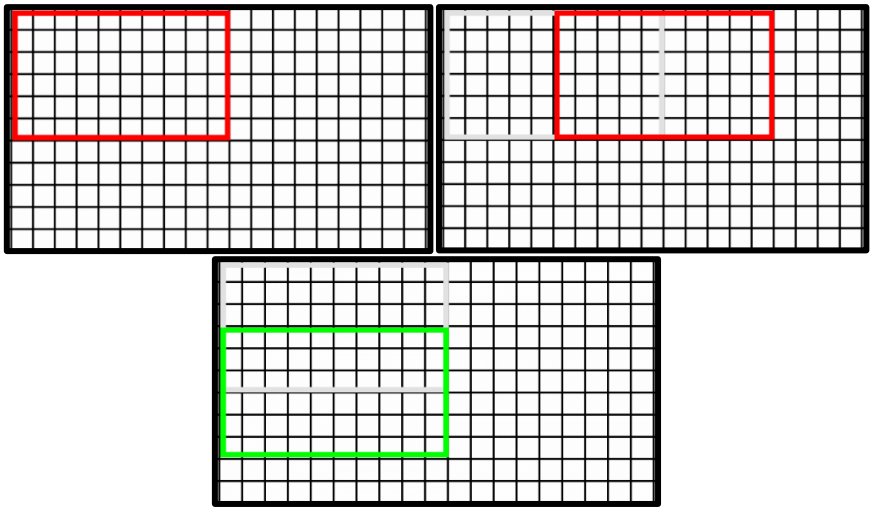
Από αυτές τις 93 εικόνες εκπαίδευσης επιλέχθηκαν τεχνικές επαύξησης δεδομένων:

1. Οι 93 εικόνες της εκπαίδευσης αντικατοπτρίστηκαν με την μέθοδο numpy.flip όπου δίνει το φαινόμενο του καθρέφτη στον οριζόντιο άξονα και δημιουργήθηκαν 93 νέες εικόνες με σύνολο:
    - 93*2 = 186 εικόνες εκπαίδευσης.
2. Για να δημιουργηθούν περισσότερες εικόνες από το υπάρχον dataset έγινε σαν δεύτερο βήμα περικοπή στις αρχικές εικόνες διαστάσεων 63x96, με κατάληξη στις διαστάσεις 42x64. Η τεχνική για να γίνει η περικοπή, έγινε με stride (βήμα pixel όπως φαίνεται και στην Εικόνα 2-14) κατάλληλο για να γίνει overlap σε κάποια pixel. Επιλέχθηκε να γίνει offline, που όπως επεξηγήθηκε και στο κεφάλαιο 0, θα αποθηκευτεί στην μνήμη του υπολογιστή σαν νέο σετ δεδομένων. Τα βήματα περιγράφονται παρακάτω:
    - Σταθερό aspect ratio: 63/96 = 42/64 = 0,65.
    - Stride Ύψους εικόνας = 7.
      Το stride αυτό υπολογίστηκε σύμφωνα με τον τύπο:
      63pixels ύψος - 42pixels τελικό ύψος (τελευταίας εικόνας) = 21 pixels
      21pixels / stride 7 = 3 εικόνες
      Άρα 4 εικόνες με ύψος της αρχικής εικόνας:
        - 0-41
        - 7-48
        - 14-56

- 21-63

- Stride πλάτους εικόνας = 8.

Με την ίδια μεθοδολογία όπως το stride ύψους με δημιουργία 5 εικόνων. Η μέθοδος περιγράφεται και στις παρακάτω εικόνες:



Εικόνα 2-14 Στην πάνω αριστερή εικόνα φαίνεται η περικοπή σε μικρότερες διαστάσεις (κόκκινο πλαίσιο). Στην πάνω δεξιά και στην κάτω εικόνα φαίνεται πως μετακινούμε αυτό το πλαίσιο αφήνοντας κάποια pixel (stride) για πλάτος (κόκκινο πλαίσιο) και ύψος (πράσινο πλαίσιο) αντίστοιχα.

3. Σαν τρίτο βήμα για να αυξηθεί ακόμα περισσότερο το σετ δεδομένων εκπαίδευσης επιλέχθηκε η αλλαγή μεγέθους των αρχικών εικόνων με διαστάσεις 63x96 σε διαστάσεις 42x64 με την μέθοδο resize της βιβλιοθήκης PIL (Python Image Library) με την εντολή PIL.Image.resize. Διαφέρει από την μέθοδο περικοπής (crop) του προηγούμενου βήματος καθώς κρατάει ολόκληρη την εικόνα.

Το τελικό dataset που προέκυψε είναι:

Εικόνες εκπαίδευσης	Εικόνες ελέγχου επίδοσης
3906	2

Οι τελικές εικόνες εκπαίδευσης προέκυψαν από:

$(93_{original} + 93_{flip}) \times 4_{CropΥψους} \times 5_{CropΠλάτους} + 186_{resized} = 3906$ εικόνες εκπαίδευσης

21x32 Dataset

Το δεύτερο πακέτο δεδομένων που χρησιμοποιήθηκε για ακόμα μεγαλύτερο σετ εκπαίδευσης και καλύτερα αποτελέσματα, είναι η περικοπή του αρχικού πακέτου σε ακόμα μικρότερες διαστάσεις με μέγεθος 21x32. Είναι το μικρότερο αρχείο που θα μπορούσε να γίνει περικοπή, κρατώντας το ίδιο aspect ratio. Τα βήματα είναι παρόμοια με τα αυτά που αναφέρθηκαν στο κεφάλαιο 42x64. Πιο συγκεκριμένα:

- Ίδιο Aspect ratio: 63/96 = 21/32 = 0,65.
  Stride Ύψους εικόνας = 7.
  Το stride αυτό υπολογίστηκε σύμφωνα με τον τύπο:
  63pixels ύψος - 21 pixels τελικό ύψος (τελευταίας εικόνας) = 42 pixels

42pixels / stride 7 = 6 εικόνες

Άρα 7 εικόνες με ύψος της αρχικής εικόνας:

- 0-21
- 7-28
- 14-35
- 21-42
- 28-49
- 35-56
- 42-63

- Stride πλάτους εικόνας = 8.

Με την ίδια μεθοδολογία όπως το stride ύψους με δημιουργία 9 εικόνων. (Η μέθοδος περιγράφεται και στην Εικόνα 2-14)

- Resize των αρχικών εικόνων και των εικόνων flip σε μέγεθος 21x32

Το τελικό dataset που χρησιμοποιήθηκε:

Εικόνες εκπαίδευσης	Εικόνες ελέγχου επίδοσης
11904	126

Και προέκυψαν από:

$(93_{original} + 93_{flip}) \times 7_{CropΥψους} \times 9_{CropΠλάτους} + 186_{resized} = 11904$ εικόνες εκπαίδευσης

$2_{εικόνεςTest} \times 7_{CropΥψους} \times 9_{CropΠλάτους} = 126$ εικόνες ελέγχου επίδοσης

21x32 dataset (b)

Η διαφορά με το προηγούμενο dataset είναι ότι, έγινε ακόμα περισσότερη επαύξηση δεδομένων πριν την περικοπή, για να δοκιμαστεί το υπάρχον δίκτυο σε όσο το δυνατόν μεγαλύτερο σετ εκπαίδευσης, για καλύτερα αποτελέσματα και για δοκιμές. Πιο συγκεκριμένα τα βήματα που ακολουθήθηκαν είναι:

- 95 αρχικές εικόνες + 95 εικόνες flip = 190 εικόνες
- 190 εικόνες x 2 = 380 εικόνες με την μέθοδο PIL.Image.rotate() για την περιστροφή των εικόνων σε 180° ώστε να κρατηθούν οι ίδιες διαστάσεις. Το αποτέλεσμα φαίνεται και στην Εικόνα 2-15
- Από αυτές 6 τυχαίες εικόνες χρησιμοποιήθηκαν για σετ ελέγχου επίδοσης (test)
- Έγινε crop με stride ύψους 7 και πλάτους 8 όπως αναφέρθηκε και στο 21x32 Dataset
- Στο σετ δεδομένων ελέγχου (test set) δεν έγινε overlap και έγινε περικοπή μόνο σε 9 εικόνες.

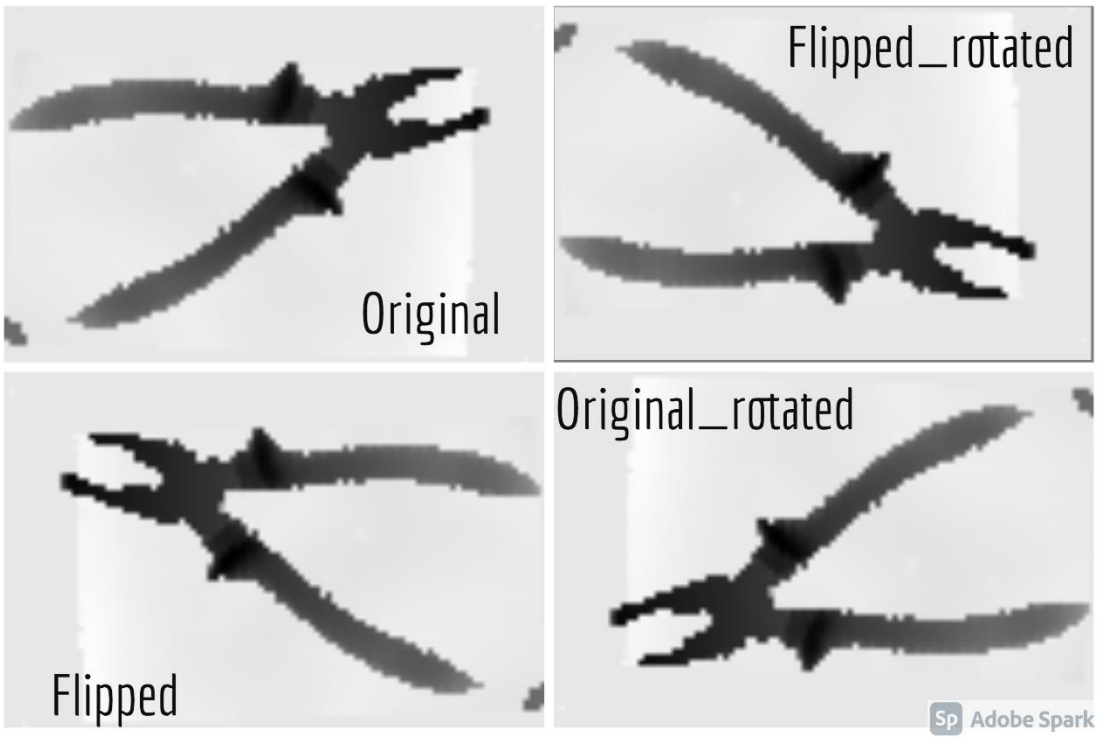
Το τελικό dataset που προέκυψε είναι:

Εικόνες εκπαίδευσης	Εικόνες ελέγχου επίδοσης
23936	54

Και προέκυψαν από:

$374_{εικόνεςTrain} \times 7_{CropΥψους} \times 9_{CropΠλάτους} + 374_{resized} = 23936$ εικόνες εκπαίδευσης

$6_{εικόνεςTest} \times 3_{CropΥψους} \times 3_{CropΠλάτους} = 54$ εικόνες Test



Εικόνα 2-15 Η επεξεργασία με αντιμετάθεση (flip) και περιστροφή (rotate) και οι εικόνες που δημιουργήθηκαν

32x32 dataset

Κατά την πορεία των προηγούμενων πακέτων δεδομένων τα αποτελέσματα δεν ήταν ιδιαίτερα ικανοποιητικά. Το 21x32 dataset (b) ήταν το μεγαλύτερο σετ δεδομένων που δοκιμάστηκε, με 23,936 εικόνες εκπαίδευσης και παρ' όλα αυτά όπως περιγράφεται και παρακάτω δεν ήταν αρκετό για να αυξήσει την ποιότητα της εικόνας εισόδου του δικτύου σύμφωνα με την εικόνα εξόδου (Ground Truth).

Για καλύτερο πειραματισμό και αποτελέσματα της αποθορυβοποίησης εικόνων βάθους, δημιουργήθηκε ακόμα ένα μεγαλύτερο πακέτο δεδομένων. Οι επιλογές που υπήρχαν από εδώ και στο εξής για περισσότερη επαύξηση δεδομένων αναλύονται παρακάτω. Η περικοπή επιλέχθηκε σε αυτές τις διαστάσεις, για να είναι εφικτό να χρησιμοποιηθεί το πακέτο δεδομένων και σε προ-εκπαιδευμένα δίκτυα όπως το VGG19[43][44] που απαιτεί ελάχιστο μέγεθος εισόδου 32x32 pixels. Σαν greyscale εικόνα βάθους, δεν δίνει δυνατότητες για επαύξηση δεδομένων με βάση την επεξεργασία στην απόχρωση.

Η μεθοδολογία που χρησιμοποιήθηκε για αυτό το dataset σε σχέση με τα προηγούμενα έχει ως εξής:

- Αφαιρέθηκαν από τις 95 αρχικές εικόνες, 16 εικόνες που θα χρησιμοποιηθούν για σετ ελέγχου επίδοσης (test set). Με 16 εικόνες μπορεί να υπολογιστεί καλύτερα η στατιστική απόδοση που θα έχει το δίκτυο στο σετ αξιολόγησης σε σχέση με τα προηγούμενα πειράματα. Στις 79 εικόνες εκπαίδευσης εφαρμόστηκαν τεχνικές επαύξησης δεδομένων.

- Η αρχική μέθοδος επαύξησης δεδομένων όπως και στα άλλα dataset περιλάμβανε καθρέφτισμα (mirror) και περιστροφή 180^{o} . Με τελικές εικόνες 79x4 = 316 εικόνες εκπαίδευσης.
- Για να καταλήξουν οι τελικές εικόνες σε μέγεθος 32x32 από το αρχικό 63x96, στις αρχικές εικόνες έγινε περικοπή στο πλάτος (δεξιά) κατά 1 px ώστε να αποκτήσουν διαστάσεις 62x96.
- Βάση αυτών των διαστάσεων αν αφαιρεθούν οι διαστάσεις 32x32:
  96- 32 = 64 pixel πλάτους
  62 – 32 = 30 pixel Ύψους
  Βάση αυτών των Pixels θα υπολογιστεί το stride που θα επιλεχθεί για την περικοπή σε μικρότερες εικόνες. Το **μικρό stride φαίνεται ότι δεν βοήθησε στα προηγούμενα dataset** γι' αυτό επιλέχθηκε μεγαλύτερο stride
  (30/6) +1 = 6 εικόνες ύψους και
  (64/8) +1 = 9 εικόνες πλάτους.
  Σύνολο = 316x54 = 17,064.
- Η λύση που ερευνήθηκε είναι να αυξηθεί το dataset μέσα από την περιστροφή των αρχικών εικόνων διαστάσεων 62x96, σε περιστροφή 90^{o} (νέες διαστάσεις 96x62). Σε αυτές τις εικόνες εφαρμόστηκε η ίδια τεχνική περικοπής με stride που αναφέρθηκε αλλά με μικρότερο stride για το νέο πλάτος. Πιο συγκεκριμένα για Stride επιλέχθηκε:
  - Stride Ύψους = 8
  - Stride Πλάτους = 5

Το τελικό dataset αποτελείται από:

Εικόνες εκπαίδευσης	Εικόνες ελέγχου επίδοσης
36972	96

Και προέκυψαν από:

$$17064_{cropped\_hrznt} + \left(316_{αρχικές\ εικόνες} * 9_{περικοπή\ ύψους} * 7_{περικοπή\ πλάτους}\right) = 36972_{εικόνες\ εκπαίδευσης}$$

$$16_{test} * 6_{cropped} = 96\ _{εικόνες\ ελέγχου}$$

Η μεθοδολογία περικοπής σε περιστροφή των αρχικών εικόνων μπορεί να αυξήσει επιπλέον το σετ δεδομένων αν η περιστροφή εφαρμοστεί σε γωνία μικρότερη των 90^{o}. Η περιστροφή σε τέτοιες γωνίες πιθανόν θα δημιουργήσει παραμόρφωση (distortion) στις αρχικές εικόνες.

2.4.Αποτελέσματα

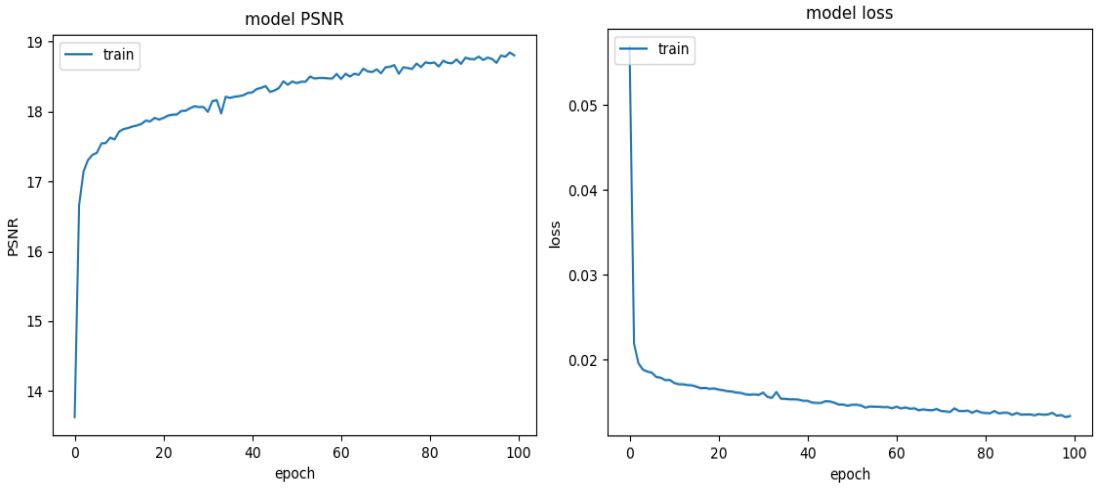
Τα αποτελέσματα που προκύπτουν από την εκτέλεση του προγράμματος στα 4 σετ δεδομένων που περιεγράφηκαν, αναλύονται στο κεφάλαιο αυτό. Η πορεία και η μεθοδολογία εξελίχθηκε έτσι ώστε, να υπάρξει καλύτερος πειραματισμός σε μικρότερα σετ δεδομένων (dataset), που θα εξάγουν χρήσιμα συμπεράσματα, ώστε στα μεγαλύτερα dataset να υπάρξει καλύτερη συνοχή των αποτελεσμάτων και καλύτερη εκτίμηση των τελικών συμπερασμάτων. Αυτό πρακτικά σημαίνει ότι με δοκιμές σε πιο μικρά σετ δεδομένων που σταδιακά θα πολλαπλασιάζονται, τα αποτελέσματα πρέπει να δείχνουν βελτίωση που θα οφείλεται και στην καλύτερη κατανόηση του προβλήματος και στην αύξηση των δεδομένων εκπαίδευσης, χωρίς μεγάλη ανάγκη για υπολογιστικό κόστος από απευθείας πειραματισμό.

- Οι χρόνοι που αναφέρονται είναι για εκτέλεση σε intel I5-8250U CPU 1.60 GHertz.
- Για το τελικό πακέτο δεδομέων που εξετάστηκαν πιο βαθιά δίκτυα χρησιμοποιήθηκε η GPU NVIDIA GTX 1050 2GB
- Η σύγκριση των αποτελεσμάτων έγινε με το PSNR και με τη μέθοδο Structural Similarity Index Measure (SSIM). Χρησιμοποιήθηκε η βιβλιοθήκη SSIM_PIL για τις μετρήσεις.
- Το PSNR που ο τύπος αναφέρθηκε στην παράγραφο 2.2 παρόλο που έχει διαφορά στα αποτελέσματα με την υλοποίηση του τύπου μέσω της numpy και με την εισαγωγή ως metrics με την βιβλιοθήκη της tensorflow [33], δίνει το απόλυτο σφάλμα στις τιμές pixels, που είναι σημαντικό για εικόνες βάθους.
- Για σύγκριση με τεχνικές επεξεργασίας εικόνων ως προς το PSNR και το SSIM με την εικόνα Ground Truth, επιλέχθηκε το φίλτρο sharpening της βιβλιοθήκης PIL.
- Το φίλτρο που αύξησε το PSNR, θεωρητικά θα βελτιώσει την δομή και την ευκρίνεια της εικόνας βάθους της RealSense που υστερεί σε ποιότητα, βελτιώνοντας το περίγραμμα των αντικειμένων.

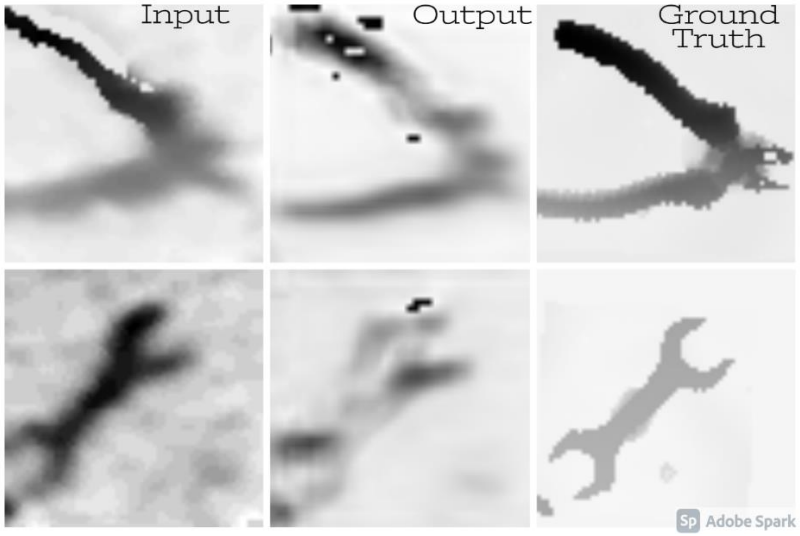
Αποτελέσματα 42x64 dataset

Για το μικρότερο dataset των 3900 εικόνων δεν γίναν αρκετά πειράματα και χρησιμοποιήθηκε κυρίως για την ανάπτυξη των αλγορίθμων και την εξοικείωση με την βιβλιοθήκη Keras και την αρχιτεκτονική των CNN.

Η αρχιτεκτονική που δοκιμάστηκε είναι με αριθμό φίλτρων 64-32-1 και μέγεθος φίλτρων 9-1-5. Τα αποτελέσματα που έδωσε το δίκτυο, δεν είναι ικανοποιητικά καθώς περιέχουν αρκετό θόρυβο, όπως φαίνεται και στην Εικόνα 2-17 και δεν κατάφεραν να προσφέρουν καλύτερα αποτελέσματα από τις εικόνες εισόδου. Τα αποτελέσματα αυτά είναι ελλιπή για εξαχθούν χρήσιμα συμπεράσματα, καθώς δεν έχει εξεταστεί το PSNR του σετ επικύρωσης (validation set), ώστε να ελεγχθεί η απόδοση του δικτύου. Παρουσιάζονται ενδεικτικά για την πορεία της εργασίας.



Εικόνα 2-16 Το PSNR αριστερά και το MSE δεξιά κατά την εκπαίδευση του δικτύου



Εικόνα 2-17 Τα αποτελέσματα του δικτύου για το dataset 42x64 που δεν κατάφερε να δώσει καλύτερα αποτελέσματα από την input εικόνα. Στην αριστερή στήλη είναι η είσοδος του δικτύου , στο κέντρο η έξοδος ενώ η hr εικόνα είναι στα δεξιά

Το PSNR και το MSE κατά το σετ ελέγχου επίδοσης:

PSNR	MSE
17.32 db	0.0124

Αποτελέσματα 21x32 dataset

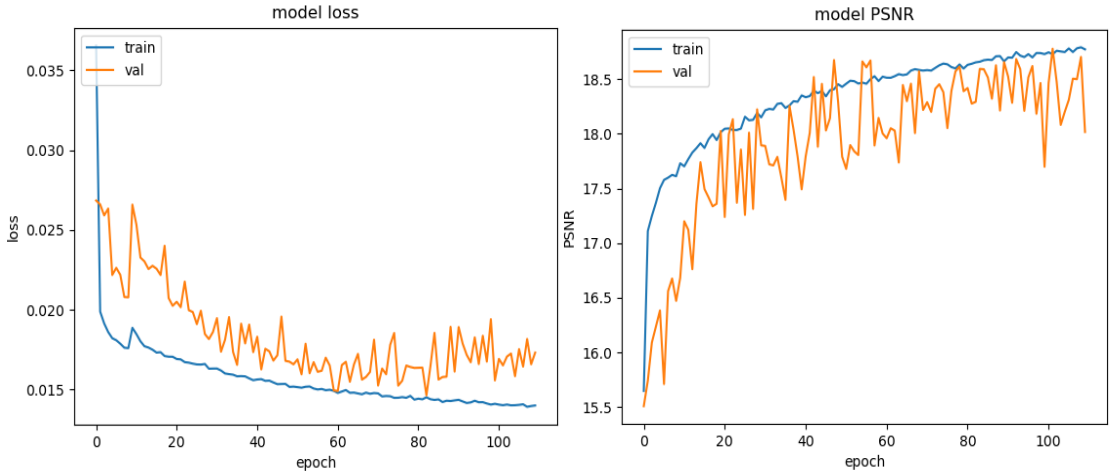
Σε αυτό το μεγαλύτερο σετ δεδομένων δοκιμάστηκαν περισσότερες αρχιτεκτονικές που δώσαν πιο ικανοποιητικά αποτελέσματα και πιο χρήσιμα συμπεράσματα. Εξετάστηκαν βασικές αρχιτεκτονικές, ώστε να επιβεβαιωθεί αν το μεγαλύτερο σετ εκπαίδευσης θα οδηγήσει σε καλύτερα αποτελέσματα. Το σετ εκπαίδευσης με 11900 εικόνες εξακολουθεί να μην είναι ικανοποιητικό για CNN.

Εμφανίζονται ενδεικτικά τα αποτελέσματα από τις δοκιμές που έγιναν, της βασικής αρχιτεκτονικής τριών στρωμάτων, με πειράματα στον αριθμό φίλτρων και στο μέγεθος των διαστάσεων τους.

Το βασικό μοντέλο που πρότεινε το CNN. Πέτυχε ικανοποιητικό δείκτη PSNR, βάση και των υπόλοιπων αρχιτεκτονικών που δοκιμάστηκαν.

Η απόδοση κατά την εκπαίδευση της αρχιτεκτονικής αυτής φαίνεται στην Εικόνα 2-18 ενώ επίσης:

- Εκπαιδεύτηκε για 110 epochs, δεν χρησιμοποιήθηκε callback
- Χρόνος για κάθε epoch 31 δευτερόλεπτα



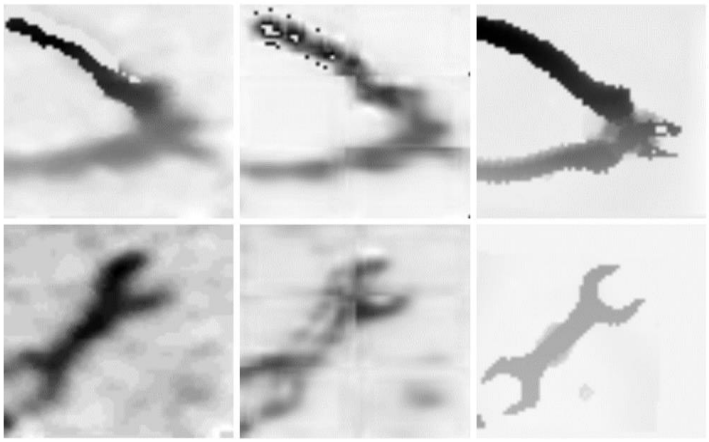
Εικόνα 2-18 Mean square error και PSNR για το σετ εκπαίδευσης και το σετ επικύρωσης (validation)

Η εκπαίδευση έδωσε το υψηλότερο PSNR στο 97^{ο} epoch που σημαίνει ότι δεν εμφάνισε υπερβολική τοποθέτηση (overfitting),ενώ με την συνάρτηση κόστους, να παρουσιάζει ένα μικρό overfitting στα δεδομένα εκπαίδευσης μετά το 82^{ο} epoch

Ο δείκτης PSNR και το MSE που πέτυχε στο σετ ελέγχου επίδοσης είναι:

PSNR	MSE
18.078 db	0.0114

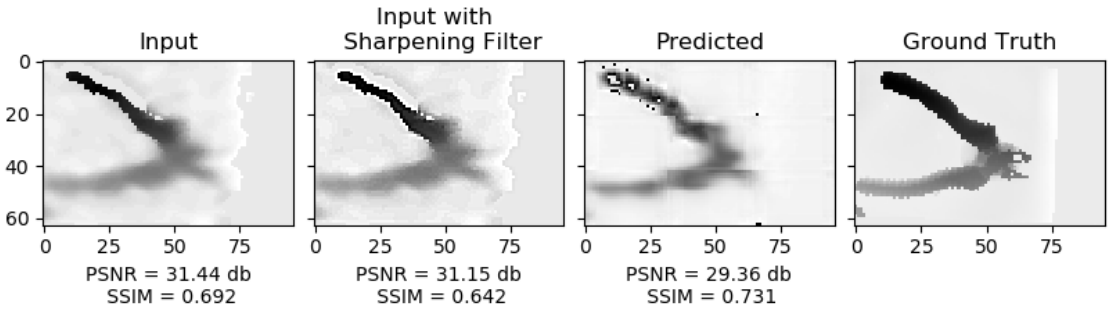
Οι έξοδοι που έδωσε το δίκτυο:



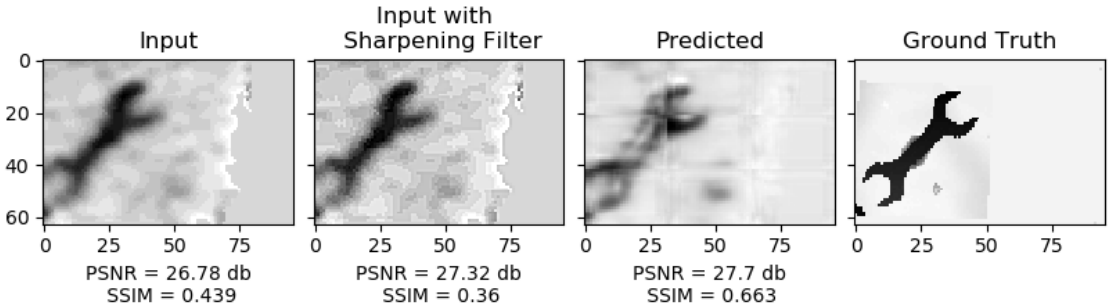
Εικόνα 2-19 Τα αποτελέσματα του δικτύου (64-32-1,9-1-5) στο σετ ελέγχου επίδοσης (μεσαίες εικόνες) σε σύγκριση με είσοδο (αριστερά) και ground truth (δεξιά).

Τα αποτελέσματα είναι σαφώς πιο βελτιωμένα από το προηγούμενο dataset που δοκιμάστηκε. Ειδικά στο «γερμανικό κλειδί», κατάφερε να βελτιώσει το βάθος και να βελτιώσει το περίγραμμα του κλειδιού. Στο «κοφτάκι» το δίκτυο δεν έδωσε ιδιαίτερα αξιόλογο αποτέλεσμα, με εμφανή θόρυβο και αστοχία πρόβλεψης του περιγράμματος και του βάθους (διαφορά απόχρωσης γκρι στα pixel).

Επίσης είναι και εμφανές το πλέγμα που προέκυψε από την επανακατασκευή της εικόνας από τα κομμάτια 21x32 Pixel. Για την επανακατασκευή ακολουθήθηκε η αντίθετη διαδικασία από την περικοπή. Δηλαδή οι αρχικές διαστάσεις 63x96 όπου εφαρμόστηκε περικοπή σε διαστάσεις 21x32 χωρίς Overlap και προέκυψαν 6 κομμάτια της αρχικής εικόνας. Τα 6 κομμάτια προστέθηκαν μεταξύ τους στο κατάλληλο ύψος και πλάτος για να ξαναδημιουργηθεί η αρχική εικόνα.



Εικόνα 2-20 Διαφορές σε PSNR και SSIM αρχιτεκτονικής 64-32-1 (κοφτάκι)



Εικόνα 2-21 Διαφορές σε PSNR και SSIM αρχιτεκτονικής 64-32-1 (γερμανικό κλειδί)

Σε αρχιτεκτονικές με μεγαλύτερο αριθμό φίλτρων το μοντέλο παρουσίασε overfitting μετά το 80ο epoch και δεν βελτίωσε σημαντικά τα αποτελέσματα. Στα επόμενα μεγαλύτερα dataset για να αποφευχθούν συνθήκες Overfitting χρησιμοποιήθηκε πρόωρο σταμάτημα (early stopping) όπως περιεγράφηκε στο κεφάλαιο 2.2 σαν μέθοδο callback της βιβλιοθήκης keras.

Αποτελέσματα 21x32 (b) dataset

Λαμβάνοντας υπόψιν την πορεία και τα αποτελέσματα των προηγούμενων πειραμάτων που δεν κατάφεραν σε ικανοποιητικό βαθμό τη βελτίωση ποιότητας (quality enhancement), το δίκτυο δοκιμάστηκε σε ένα ακόμα μεγαλύτερο dataset.

- Εδώ το σετ επικύρωσης (Validation set) είναι το 5% των εικόνων εκπαίδευσης.

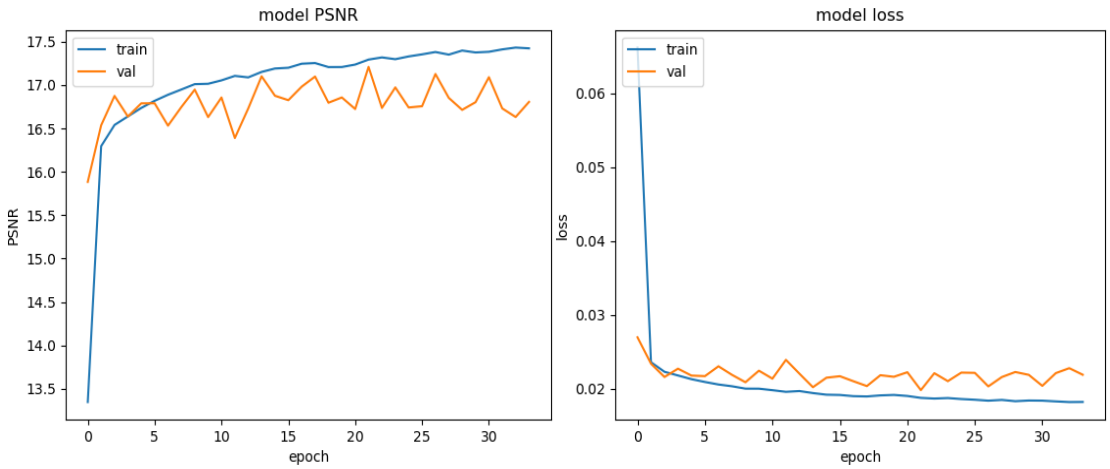
- Χρησιμοποιήθηκε early stopping στα που ελέγχει αν για 10 epochs δεν υπάρχει μείωση στο validation MSE.

Για την πρώτη αρχιτεκτονική παρουσιάζονται αναλυτικά τα αποτελέσματα για κάθε εικόνα σε PSNR και SSIM καθώς και στα επόμενα πειράματα ακολουθεί το ίδιο μοτίβο με το ποσοστό επιτυχίας αποθορυβοποίησης για κάθε αντικείμενο.

Αρχιτεκτονική 64-32-1, 9-1-5

Για το βασικό μοντέλο:

- Ελάχιστο MSE στο validation set: 0.0198
- Μέγιστο PSNR στο validation set: 17.209 db
- Χρόνος εκτέλεσης: 19 λεπτά και 4 δευτερόλεπτα.

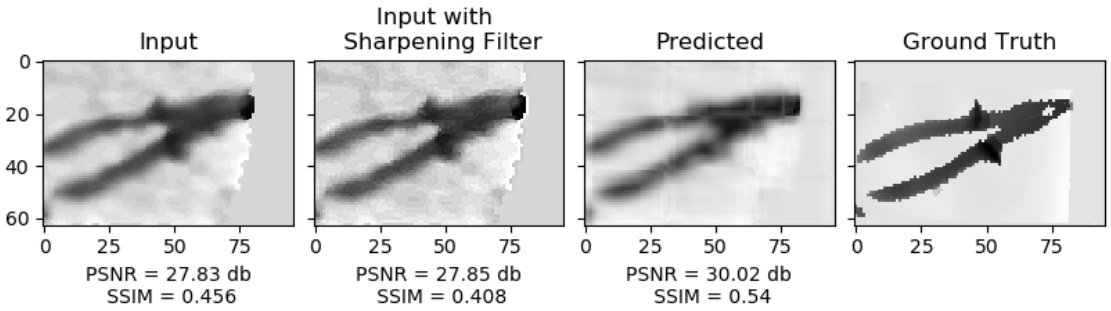


Εικόνα 2-22 Το PSNR και το MSE κατά την εκπαίδευση. Με μπλε χρώμα το σετ εκπαίδευσης και με πορτοκαλί το σετ επικύρωσης (training set,validation set).

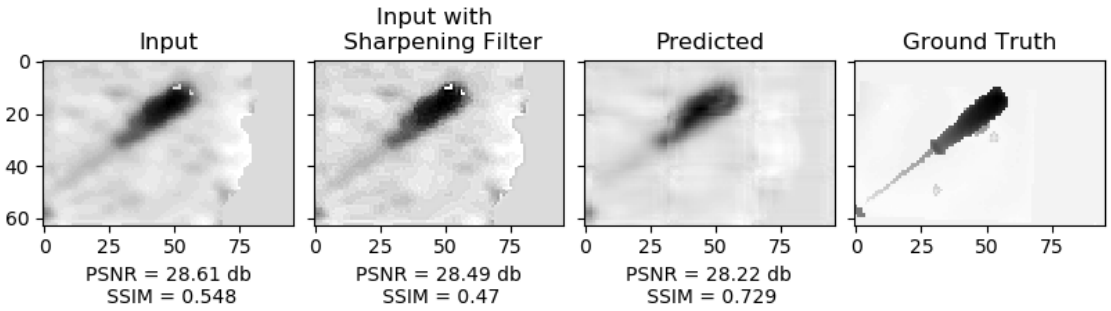
Αποτελέσματα στο σετ ελέγχου επίδοσης είναι:

MSE	PSNR
0.0178	17.698 db

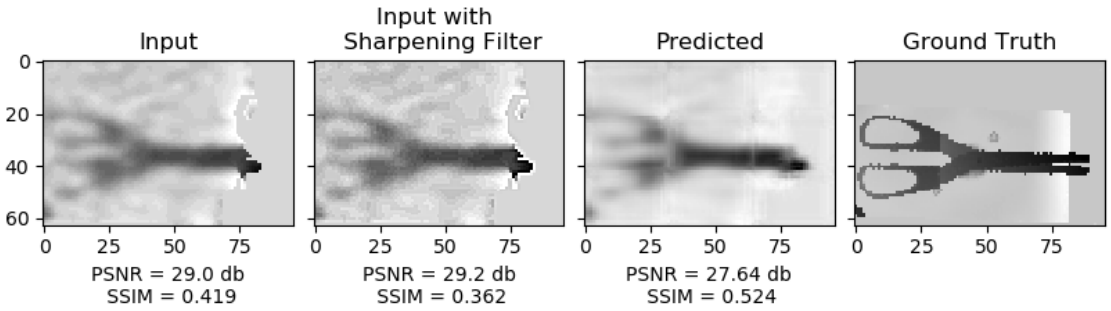
Αναλυτικά παρουσιάζονται οι έξοδοι του δικτύου για αυτήν την αρχιτεκτονική με τις τιμές PSNR και SSIM, σε σύγκριση με την εικόνα ground truth στις παρακάτω εικόνες.



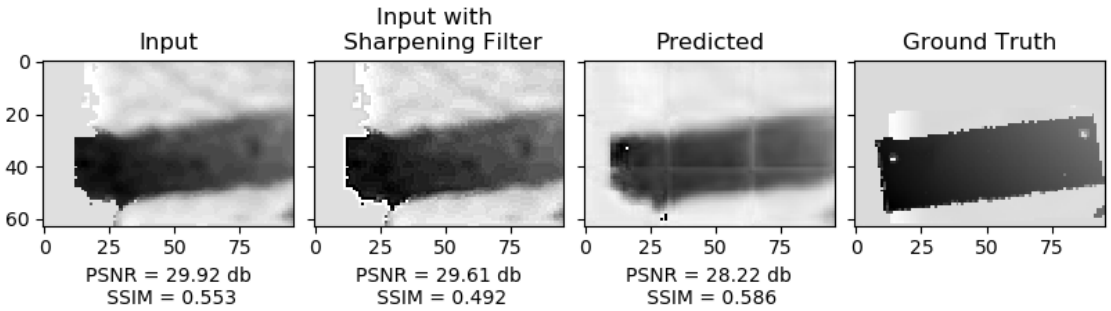
Εικόνα 2-23 Αποτελέσματα (πένσα)



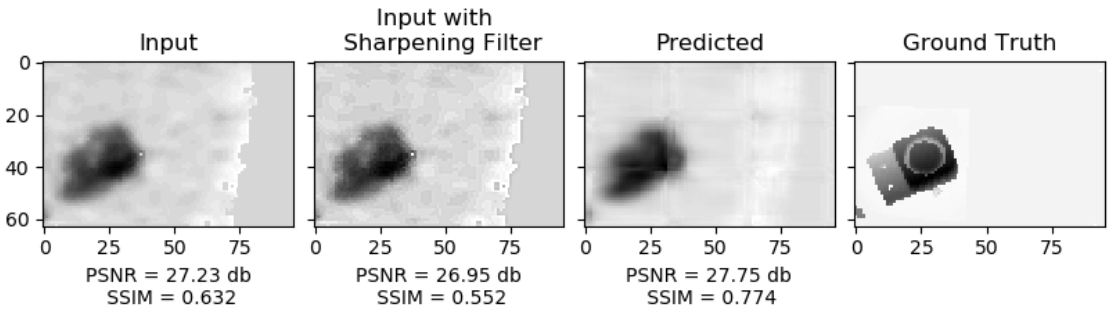
Εικόνα 2-24 Αποτελέσματα (κατσαβίδι)



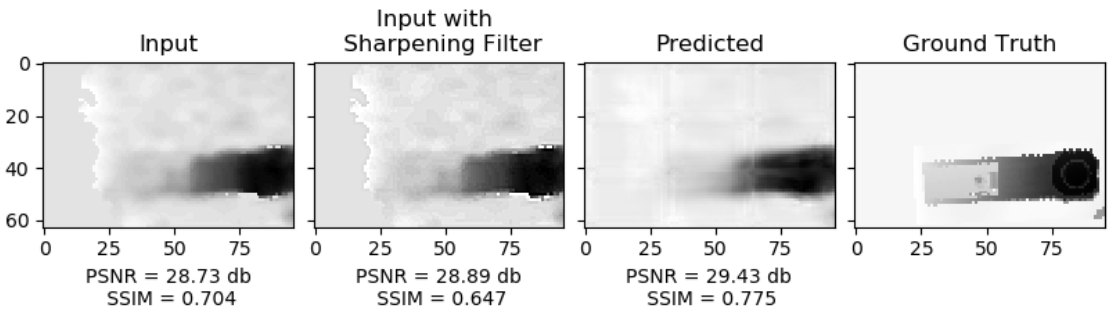
Εικόνα 2-25 Αποτελέσματα (Ψαλίδι)



Εικόνα 2-26 Αποτελέσματα (Χάρακας)



Εικόνα 2-27 Αποτελέσματα (USB)



Εικόνα 2-28 Αποτελέσματα (USB2)

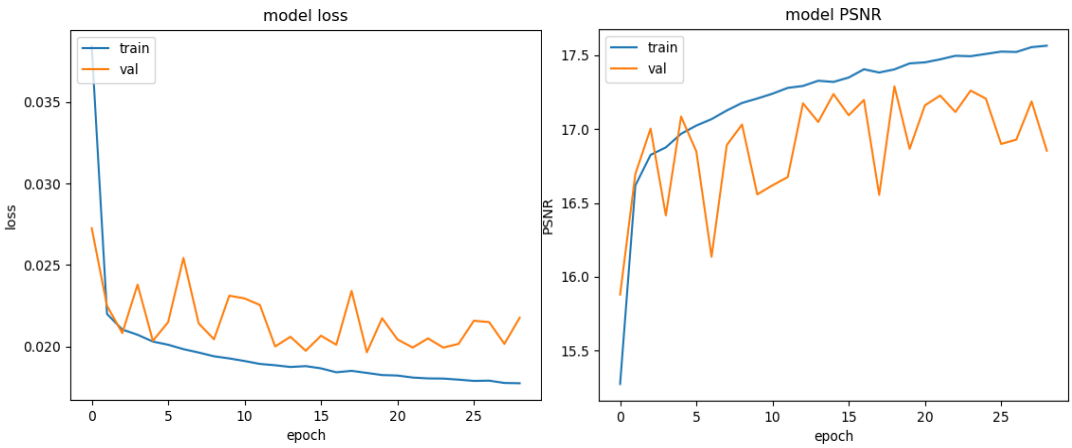
Οι εικόνες που έδωσε το δίκτυο στην έξοδο του βάση των παραπάνω εικόνων, δεν είναι ιδιαίτερα ικανοποιητικές.

- Το SSIM βελτιώθηκε και για τις 6 εικόνες
- Το PSNR της εικόνας του δικτύου ήταν καλύτερο σε 3 εικόνες από τις 6.
- Φαίνεται μια βελτίωση στον γενικό θόρυβο που αποδεικνύεται και από την αύξηση του SSIM.
- Στο περίγραμμα, στις επιφάνειες με μικρό πλάτος όπως το κατσαβίδι χειροτέρεψε το αποτέλεσμα, καθώς πιθανόν αναγνωρίστηκε ως θόρυβος και αυτό φαίνεται σε σύγκριση και με τα αποτελέσματα PSNR.
- Το ίδιο συνέβη και σε ακμές όπως τα χερούλια της πένσας που είναι μειωμένες βάση της εισόδου.
- Στο βάθος (διαβάθμιση του γκρι στα pixels) φαίνεται μια μικρή βελτίωση.

Αρχιτεκτονική 128-64-1, 9-1-5

Για το πρώτο μοντέλο με μεγαλύτερο αριθμό φίλτρων:

- Ελάχιστο MSE στο validation set: 0.02
- Μέγιστο PSNR στο validation set: 17.18 db
- Χρόνος εκτέλεσης: 31 λεπτά και 30 δευτερόλεπτα.



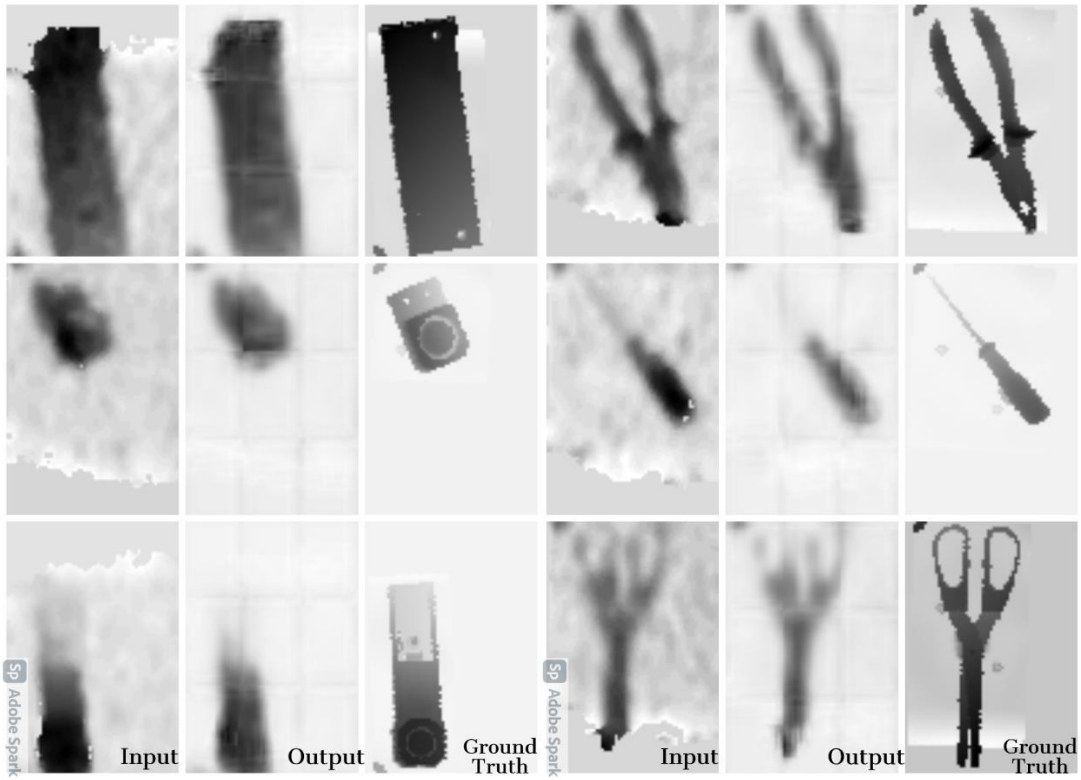
Εικόνα 2-29 MSE και PSNR στο σετ εκπαίδευσης και επικύρωσης (training set, validation set)

Αποτελέσματα στο σετ ελέγχου επίδοσης είναι:

MSE	PSNR
0.0187	17.6254 db

Βάση των αποτελεσμάτων της εκπαίδευσης το δίκτυο αυτό είναι χειρότερο από το 64-32-1.

Οι εικόνες που έδωσε στην έξοδο φαίνονται παρακάτω:

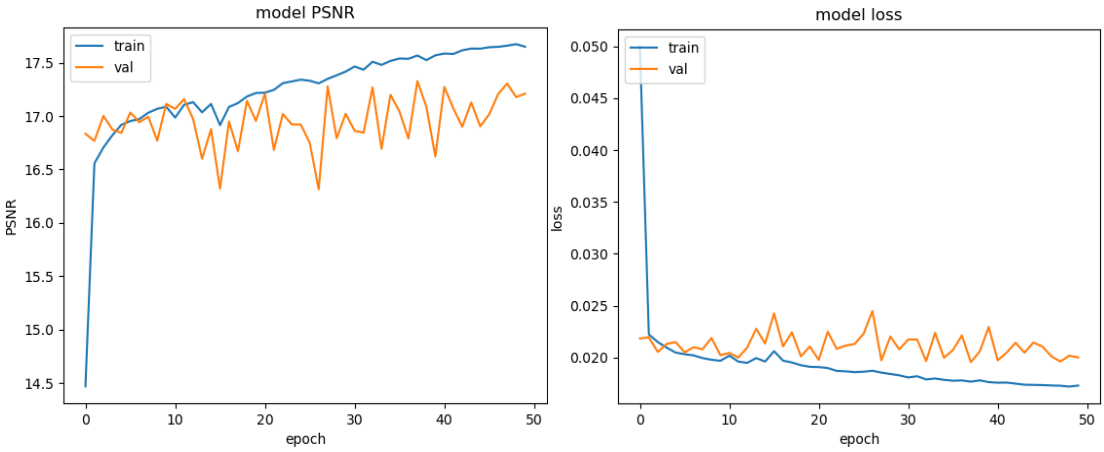


Εικόνα 2-30 Αποτελέσματα δικτύου (128-64-1, 9-1-5)

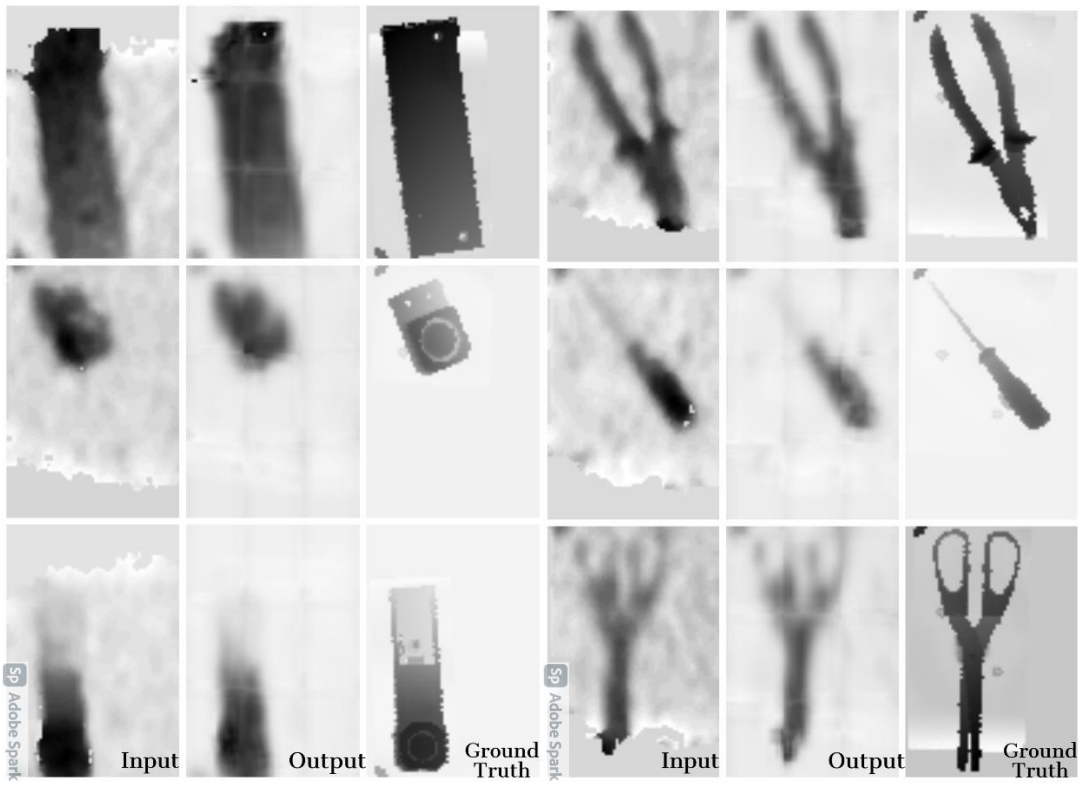
Η αρχιτεκτονική δεν έδωσε ικανοποιητικά αποτελέσματα. Με τις γωνίες του περιγράμματος και το βάθος να τείνουν να εξαλειφθούν.

Αρχιτεκτονική 256-128-1, 9-1-5

- Ελάχιστο MSE στο validation set: 0.02
- Μέγιστο PSNR στο validation set: 17.22 db
- Χρόνος εκτέλεσης: 1 ώρα 47 λεπτά και 35 δευτερόλεπτα.



Εικόνα 2-31 PSNR και MSE στο σετ εκπαίδευσης και επικύρωσης (training set, validation set)



Εικόνα 2-32 Αποτελέσματα αρχιτεκτονικής 256-128-1

Αποτελέσματα στο σετ ελέγχου επίδοσης είναι:

MSE	PSNR
0.0175	17.873 db

Και σε αυτήν την αρχιτεκτονική αν και το PSNR που πέτυχε ήταν υψηλότερο από τις προηγούμενες αρχιτεκτονικές, τα αποτελέσματα δεν είναι ικανοποιητικά.

- Τείνει να εξαλείψει τις άκρες
- Το βάθος βελτιώθηκε στην λαβή του «κατσαβιδιού» και στο «usb».

Αρχιτεκτονική 64-32-1, 11-1-7

Έχοντας το καλύτερο αποτέλεσμα στις εικόνες εξόδου, δοκιμάστηκε και για μεγαλύτερο μέγεθος kernel:

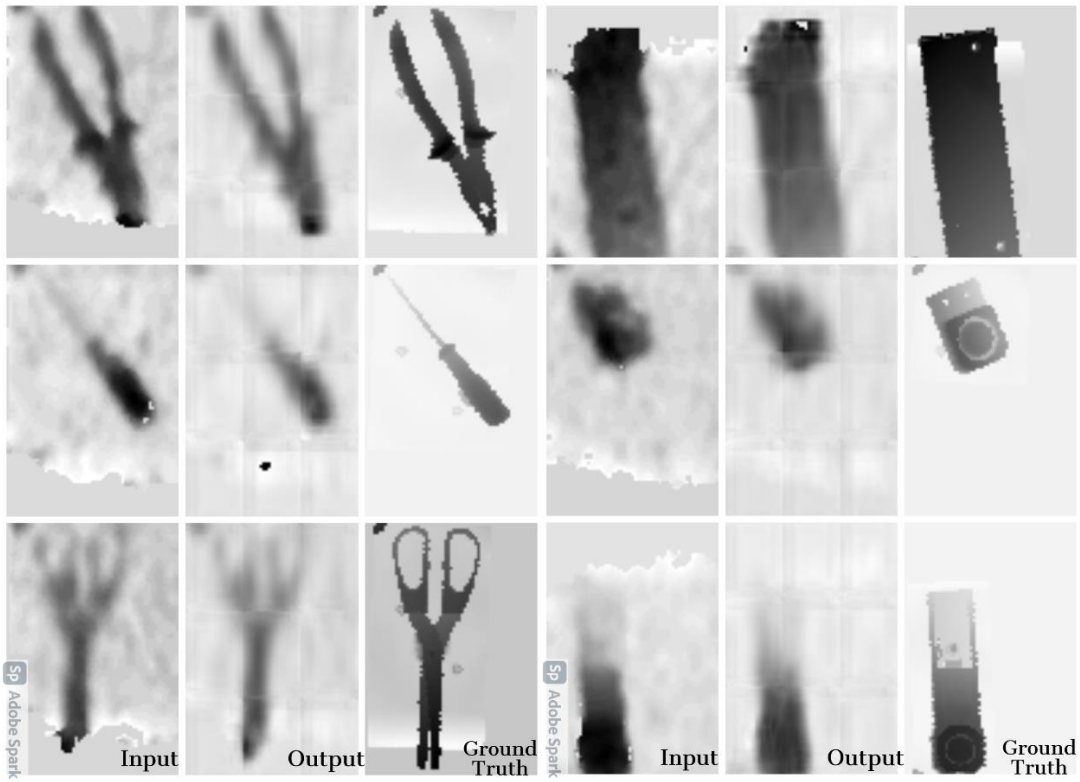
- Ελάχιστο MSE στο validation set: 0.0205
- Μέγιστο PSNR στο validation set: 17.035 db
- Χρόνος εκτέλεσης: 28 λεπτά και 37 δευτερόλεπτα.

Οι παραπάνω τιμές είναι ελαφρώς χειρότερες από αυτές με μικρότερο μέγεθος kernel της πρώτης αρχιτεκτονικής.

Αποτελέσματα στο σετ ελέγχου επίδοσης είναι:

MSE	**PSNR**
0.018	17.611 db

Βάση των MSE και του PSNR του σετ ελέγχου επίδοσης δεν υπάρχει μεγάλη διαφορά με το πρώτο μοντέλο με μικρότερο μέγεθος kernel. Οι εικόνες εξόδου του δικτύου φαίνονται παρακάτω:



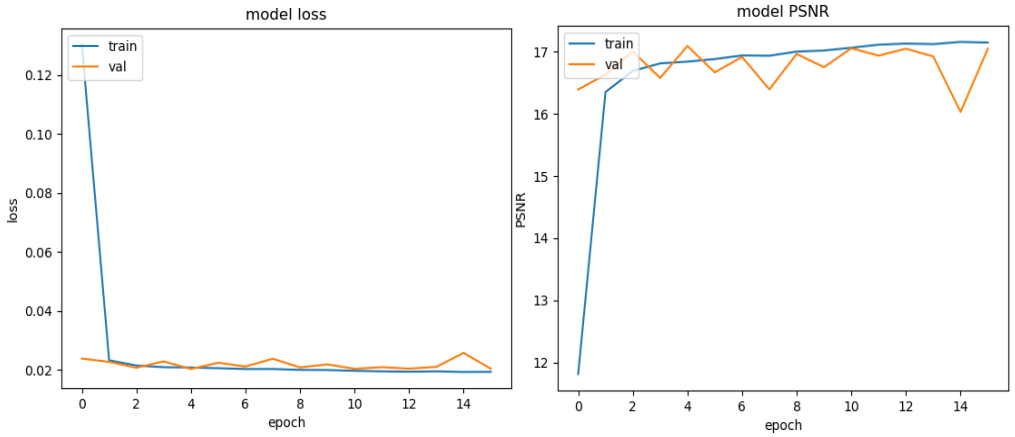
Εικόνα 2-33 Αποτελέσματα αρχιτεκτονικής 64-32-1, 11-1-7

Τα αποτελέσματα δεν είναι ικανοποιητικά και δεν είναι καλύτερα από την Αρχιτεκτονική 64-32-1, 9-1-5 το μεγαλύτερο μέγεθος δεν φαίνεται να απέδωσε.

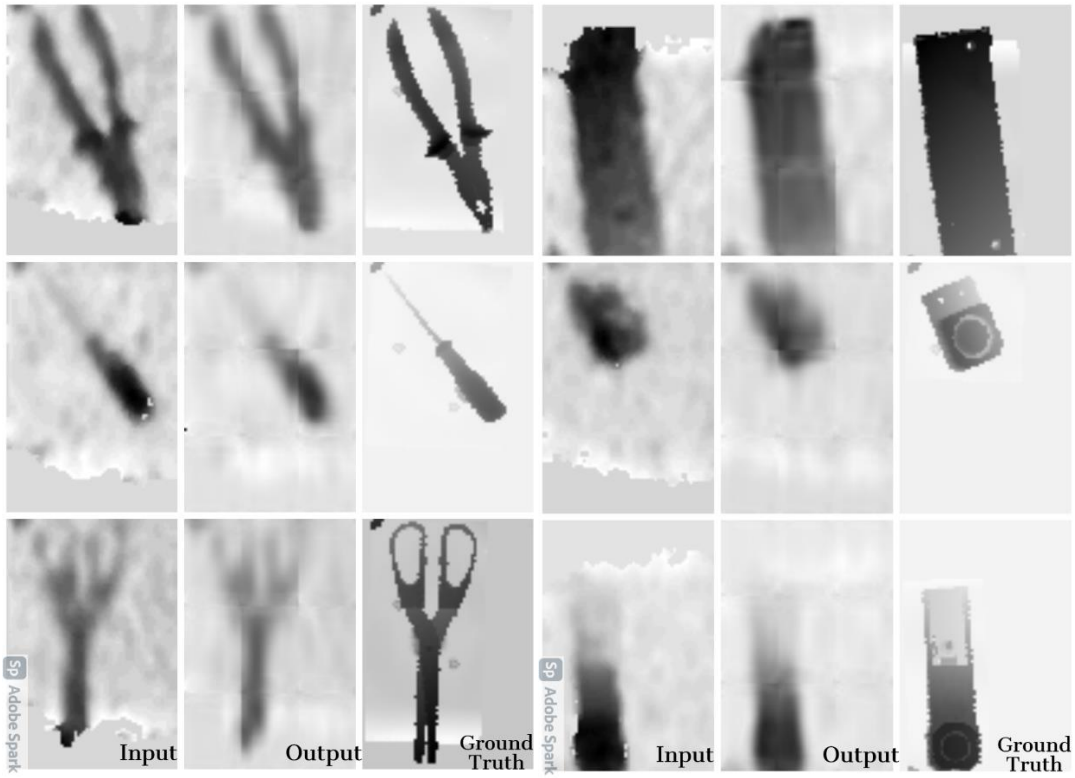
Αρχιτεκτονική 128-64-1, 11-5-7

Έχοντας το χειρότερο αποτέλεσμα στις εικόνες εξόδου, δοκιμάστηκε και για μεγαλύτερο μέγεθος kernel:

- Ελάχιστο MSE στο validation set: 0.0203
- Μέγιστο PSNR στο validation set: 17.09 db
- Χρόνος εκτέλεσης: 67 λεπτά και 25 δευτερόλεπτα.



Εικόνα 2-34 MSE και PSNR στο σετ εκπαίδευσης και επικύρωσης (training set, validation set)



Εικόνα 2-35 Αποτελέσματα αρχιτεκτονικής 128-64-1, 11-5-7

Αποτελέσματα στο σετ ελέγχου επίδοσης είναι:

MSE	PSNR
0.0183	17.51 db

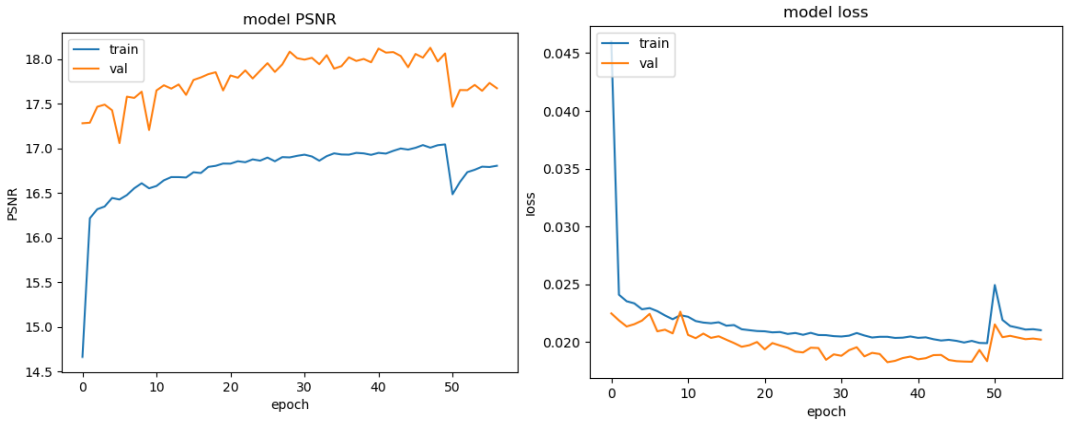
Σαν αρχιτεκτονική η 128-64-1 δεν απέδωσε με μικρότερο μέγεθος kernel με μέχρι στιγμής τα χειρότερα αποτελέσματα. Με μεγαλύτερο μέγεθος kernel σε μερικές από τις εικόνες κατάφερε καλύτερα αποτελέσματα από τις εικόνες εισόδου σε βάθος και περίγραμμα.

Αποτελέσματα τελικού dataset 32x32

Βάση των προηγούμενων αποτελεσμάτων, από τις δοκιμές στα προηγούμενα πακέτα δεδομένων και της μεθοδολογίας που ακολουθήθηκε, υπήρχαν αρκετά περιθώρια βελτίωσης που έπρεπε να εξεταστούν. Σε αυτό το dataset παρουσιάζονται τα τελικά αποτελέσματα που πέτυχαν σε υψηλό βαθμό την αποθορυβοποίηση των εικόνων βάθους χαμηλής ποιότητας.

Για το μεγαλύτερο και τελικό σετ δεδομένων έγιναν οι περισσότερες δοκιμές. Πέρα από τις αρχιτεκτονικές του SRCNN που δοκιμάστηκαν στα προηγούμενα σετ, δοκιμάστηκαν και πιο βαθιά δίκτυα για να συγκριθεί το αποτέλεσμα τους με τα αποτελέσματα του SRCNN για το συγκεκριμένο πρόβλημα.

- Οι χρόνοι εκπαίδευσης διαφέρουν κατά πολύ με την χρήση GPU και βοήθησε πολύ στον πειραματισμό με βαθιά δίκτυα
- Η μεθοδολογία αυτή ήταν πιο ικανοποιητική και φάνηκε και από τα γραφήματα κατά την εκπαίδευση όπου το dataset αυτό χρειάστηκε περισσότερα epochs σε σχέση με τα προηγούμενα dataset για την εκπαίδευσή του, χωρίς να παρουσιάσει νωρίς υπερβολική τοποθέτηση (overfitting). Ενδεικτικά παρουσιάζονται γραφήματα από την αρχιτεκτονική του βασικού μοντέλου.



Εικόνα 2-36 Αρχιτεκτονική 64-32-1 PSNR και MSE κατά την εκπαίδευση

Συγκεντρωτικά αποτελέσματα

Εδώ εμφανίζονται τα συγκεντρωτικά αποτελέσματα όπου συγκρίνονται οι περικομμένες εικόνες των αρχικών του σετ ελέγχου επίδοσης με αυτές που έδωσε στην έξοδο το δίκτυο. Οι εικόνες είναι 96 και προκύπτουν ως 6 περικομμένες εικόνες για κάθε μία από τις αρχικές

με 6*16 = 96. Η σύγκριση έγινε με το PSNR και το SSIM της εικόνας εισόδου, της εικόνας εισόδου με φίλτρο sharpening και της εικόνας Ground Truth.

Αρχιτεκτονική	Αναλογία καλύτερου PSNR	Αναλογία καλύτερου SSIM	Ποσοστό απόδοσης δικτύου % PSNR	Ποσοστό απόδοσης δικτύου % SSIM
64-32-1	60/96	75/96	62.5	78.12
128-64-1	79/96	71/96	82.29	73.95
128-64-32-1	77/96	74/96	80.2	77.08
256-128-1	72/96	76/96	75	79.16
Simple 128-1	60/96	70/96	62.5	72.9
64-32-1 skip connection	73/96	32/96	76.04	33.34
128-64-1 skip connection	77/96	71/96	80.2	73.95
256-128-1 skip connection	66/96	74/96	68.75	77.08
Δίκτυο 8 στρωμάτων (Εικόνα 2-11)	74/96	65/96	77.08	67.70
Δίκτυο 19 στρωμάτων	79/96	60/96	82.29	62.5
ESRCNN	70/96	70/96	72.91	72.91
Encoder (Εικόνα 2-12)	71/96	68/96	73.95	70.83

Πίνακας 2-2 Πίνακας συγκεντρωτικών αποτελεσμάτων με σύγκριση των εικόνων εξόδου, των αρχιτεκτονικών που δοκιμάστηκαν γι' αυτό το dataset. Με πράσινο χρώμα οι καλύτερες επιδόσεις

Οι εικόνες εξόδου συγκρίθηκαν με το PSNR των εικόνων εισόδου, και των εικόνων εισόδου με φίλτρο Sharpening. Η αναλογία καλύτερου αποτελέσματος λήφθηκε υπόψιν βάση της συγκρίσεως και των δύο και μόνο αν η predicted εικόνα είχε καλύτερα αποτελέσματα και από τις δύο θεωρήθηκε καλύτερο αποτέλεσμα.

Τα αποτελέσματα στην επανακατασκευή των εικόνων σε διαστάσεις 62x96 pixels από τις 6 περικομμένες εικόνες 21x32 pixels, για τις 3 καλύτερες αρχιτεκτονικές του Πίνακας 2-2 φαίνονται στους παρακάτω πίνακες:

Αρχιτεκτονική	Αναλογία καλύτερου PSNR reconstructed	Αναλογία καλύτερου SSIM reconstructed	Ποσοστό επιτυχίας PSNR %	Ποσοστό επιτυχίας SSIM %
128-64-1	14/14	14/14	87.5	87.5
128-64-32-1	14/16	15/16	87.5	93.75
256-128-1	14/16	15/16	87.5	93.75

Πίνακας 2-3 Σύγκριση αποτελεσμάτων για τις reconstructed εικόνες, με σύγκριση των εικόνων εξόδου, των αρχιτεκτονικών που δοκιμάστηκαν γι' αυτό το dataset. Με πράσινο χρώμα οι καλύτερες επιδόσεις

Στους παρακάτω πίνακες φαίνεται αναλυτικά τα αποτελέσματα για κάθε έξοδο (reconstructed) των καλύτερων δικτύων του πίνακα 2-3.

ΑΡΧΙΤΕΚΤΟΝΙΚΗ 128_64_32_1

ΕΙΚΟΝΑ	Input PSNR db	Input with sharpening filter PSNR db	**Predicted PSNR db**	Input SSIM	Input with sharpening filter SSIM	**Predicted SSIM**
1	27.706	27.196	**29.605**	0.729	0.6685	**0.8486**
2	27.214	27.437	**33.794**	0.647	0.573	**0.8294**
3	27.37	27.372	**30.088**	0.5359	0.4772	**0.6908**
4	27.926	27.834	**30.931**	0.6286	0.574	**0.6864**
5	27.3	27.38	**35.425**	0.7534	0.6842	**0.9025**
6	26.534	27.254	**30.471**	0.651	0.5955	**0.7503**
7	28.026	28.403	**28.822**	0.6986	0.645	0.6973
8	28.012	27.57	**28.613**	0.6851	0.6316	**0.7885**
9	28.642	28.712	**32.277**	0.68	0.6212	**0.8084**
10	30.8	30.583	29.999	0.7624	0.7244	**0.7657**
11	28.733	28.566	27.707	0.3925	0.3462	**0.4054**
12	26.68	27.183	**29.35**	0.7355	0.6815	**0.8119**
13	26.688	27.171	**29.971**	0.7932	0.7382	**0.8296**
14	26.81	26.974	**30.887**	0.5936	0.5121	**0.7728**
15	28.676	28.81	**32.562**	0.662	0.6031	**0.7857**
16	28.351	28.675	**31.996**	0.6981	0.6438	**0.7777**

Πίνακας 2-4 Τα αποτελέσματα σε PSNR και SSIM της αρχιτεκτονικής 128_64_32_1

ΑΡΧΙΤΕΚΤΟΝΙΚΗ 256_128_1

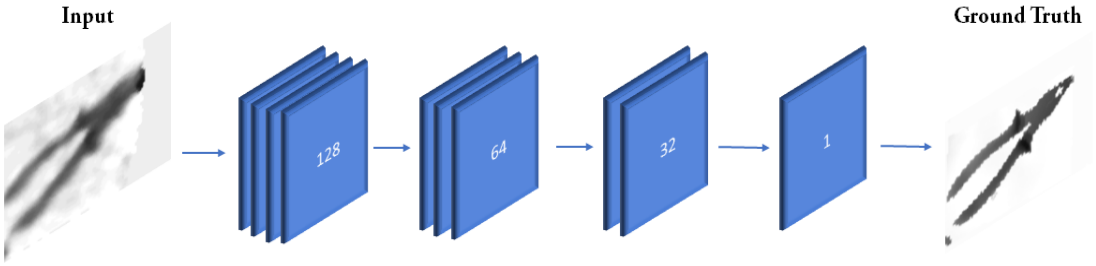
ΕΙΚΟΝΑ	Input PSNR db	Input with sharpening filter PSNR db	**Predicted PSNR db**	Input SSIM	Input with sharpening filter SSIM	**Predicted SSIM**
1	27.706	27.196	**29.398**	0.729	0.6685	**0.829**
2	27.214	27.437	**32.748**	0.647	0.573	**0.8301**
3	27.37	27.372	**29.56**	0.5359	0.4772	**0.7096**
4	27.926	27.834	**31.426**	0.6286	0.574	**0.6993**
5	27.3	27.38	**34.96**	0.7534	0.6842	**0.8981**
6	26.534	27.254	**31.642**	0.651	0.5955	**0.7429**
7	28.026	28.403	**29.579**	0.6986	0.645	0.697
8	28.012	27.57	**29.492**	0.6851	0.6316	**0.7777**
9	28.642	28.712	**32.951**	0.68	0.6212	**0.7871**
10	30.8	30.583	28.408	0.7624	0.7244	**0.7649**
11	28.733	28.566	27.099	0.3925	0.3462	**0.4413**
12	26.68	27.183	**30.462**	0.7355	0.6815	**0.8124**
13	26.688	27.171	**30.434**	0.7932	0.7382	**0.8479**
14	26.81	26.974	**29.349**	0.5936	0.5121	**0.7843**
15	28.676	28.81	**32.11**	0.662	0.6031	**0.7827**
16	28.351	28.675	**32.761**	0.6981	0.6438	**0.7879**

Πίνακας 2-5 Τα αποτελέσματα σε PSNR και SSIM της αρχιτεκτονικής 256_129_1

Οι καλύτερες αρχιτεκτονικές του **Error! Reference source not found.** και τα αποτελέσματα τους αναλύονται παρακάτω.

Αρχιτεκτονική 128-64-32-1

Καλύτερη αρχιτεκτονική βάση του Πίνακας 2-2 είναι η αρχιτεκτονική 4 στρωμάτων με αριθμό φίλτρων 128-64-32-1 και μέγεθος φίλτρων 11-5-3-5. Όπου συνδύασε αρκετά υψηλό PSNR και SSIM. Η αρχιτεκτονική του φαίνεται και στην παρακάτω εικόνα.



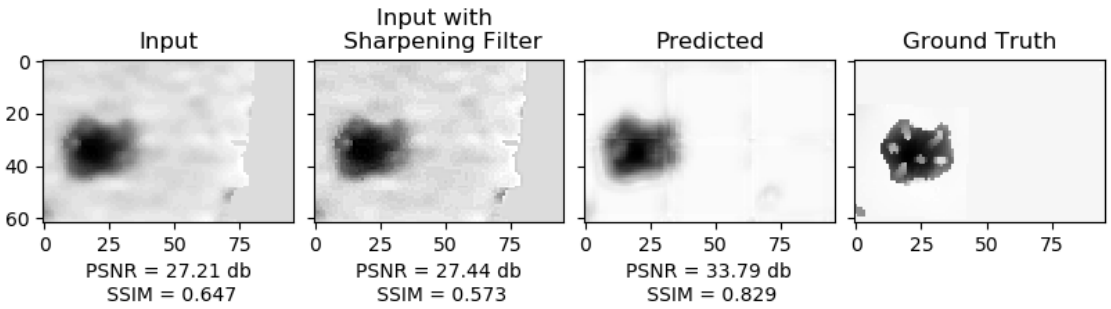
Εικόνα 2-37 Αρχιτεκτονική 128-64-32-1 με μέγεθος φίλτρων 11-5-3-1

Κατά την εκπαίδευση:

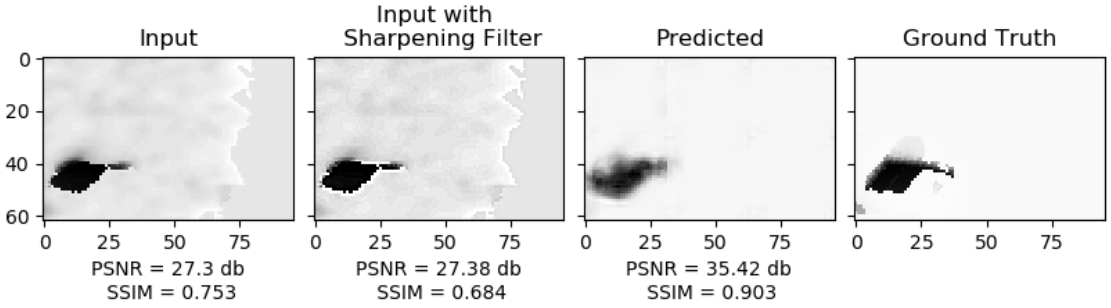
- Εκπαιδεύτηκε για 60 epochs με χρόνο εκπαίδευσης 17 λεπτά, και 13 δευτερόλεπτα
- Ελάχιστο Mean Square Error στο validation set: 0.0183
- Μέγιστο PSNR στο validation set: 18.127 db

Στο σετ ελέγχου επίδοσης: MSE 0.0181, PSNR 17.42 db

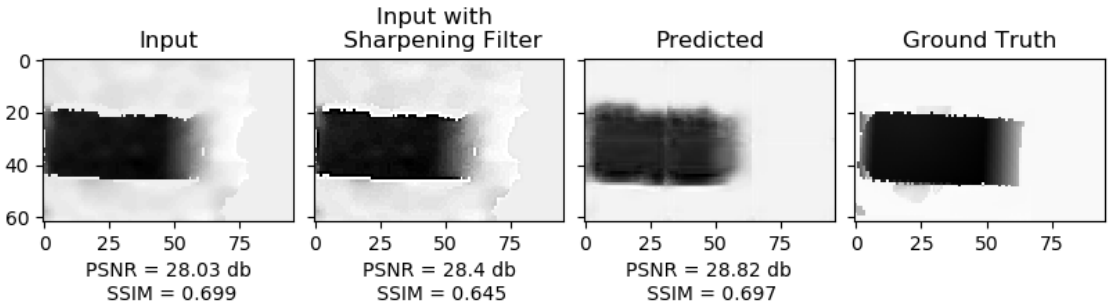
Στις παρακάτω εικόνες παρουσιάζονται μερικά από τα αποτελέσματα των επανα-κατασκευασμένων εικόνων βάση του **Error! Reference source not found.**:



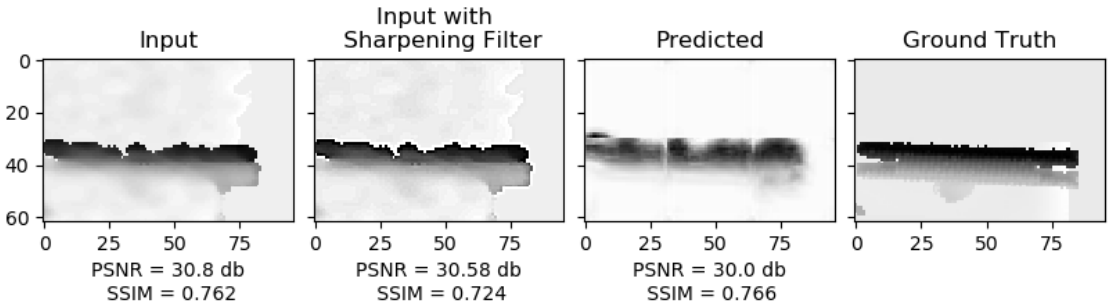
Εικόνα 2-38 Η έξοδος του δικτύου που πέτυχε το μεγαλύτερο SSIM και αρκετά ψηλό PSNR βάση του σετ ελέγχου επίδοσης.



Εικόνα 2-39 Η έξοδος με το μεγαλύτερο PSNR του δικτύου βάση του σετ ελέγχου επίδοσης.



Εικόνα 2-40 Η έξοδος με το χειρότερο SSIM βάση του σετ ελέγχου επίδοσης.



Εικόνα 2-41 Η έξοδος με το χειρότερο PSNR του σετ ελέγχου επίδοσης.

Αρχιτεκτονική 256-128-1

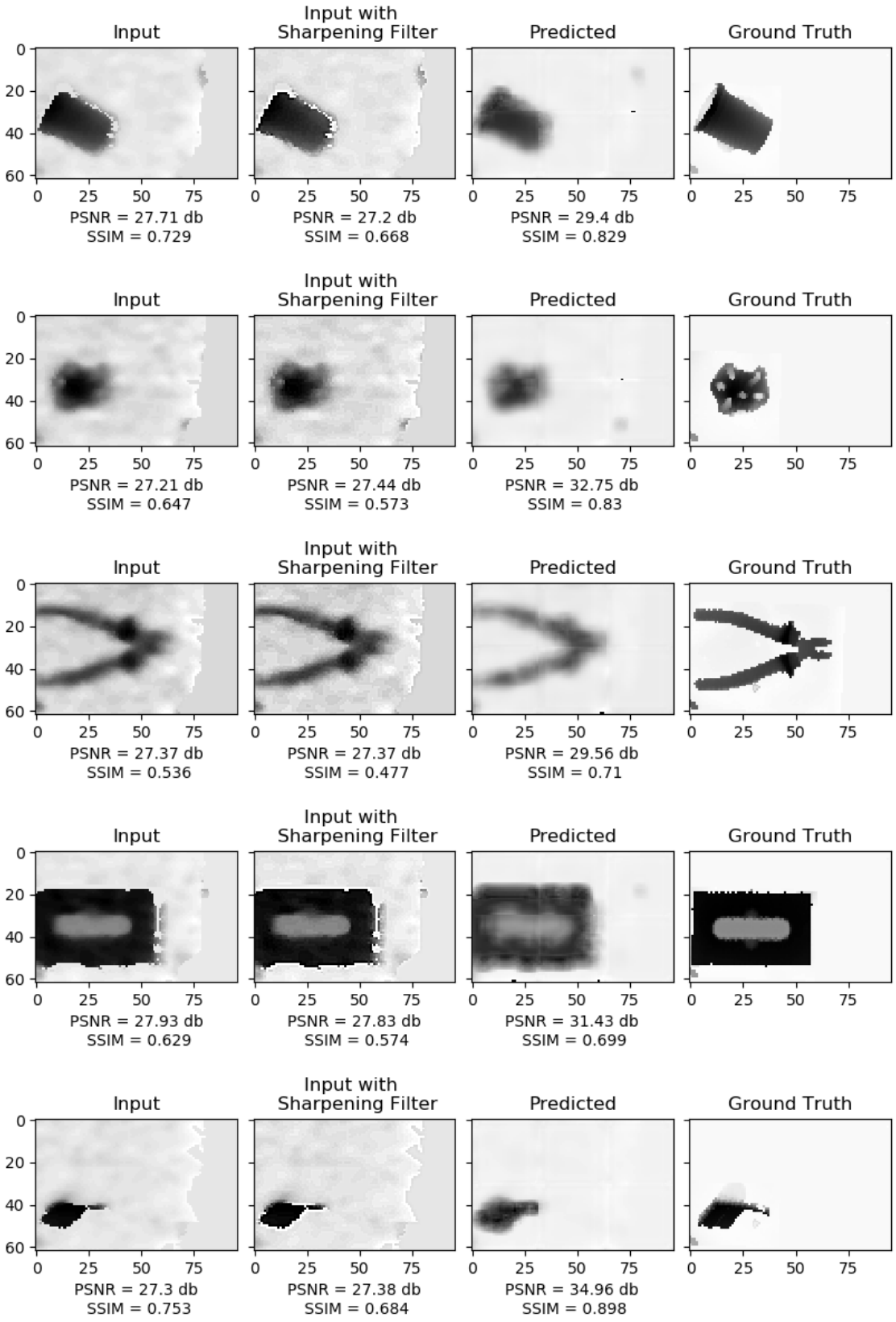
Αρχιτεκτονική δικτύου με μέγεθος φίλτρων 9-3-5,και με αριθμό φίλτρων 256-128-1

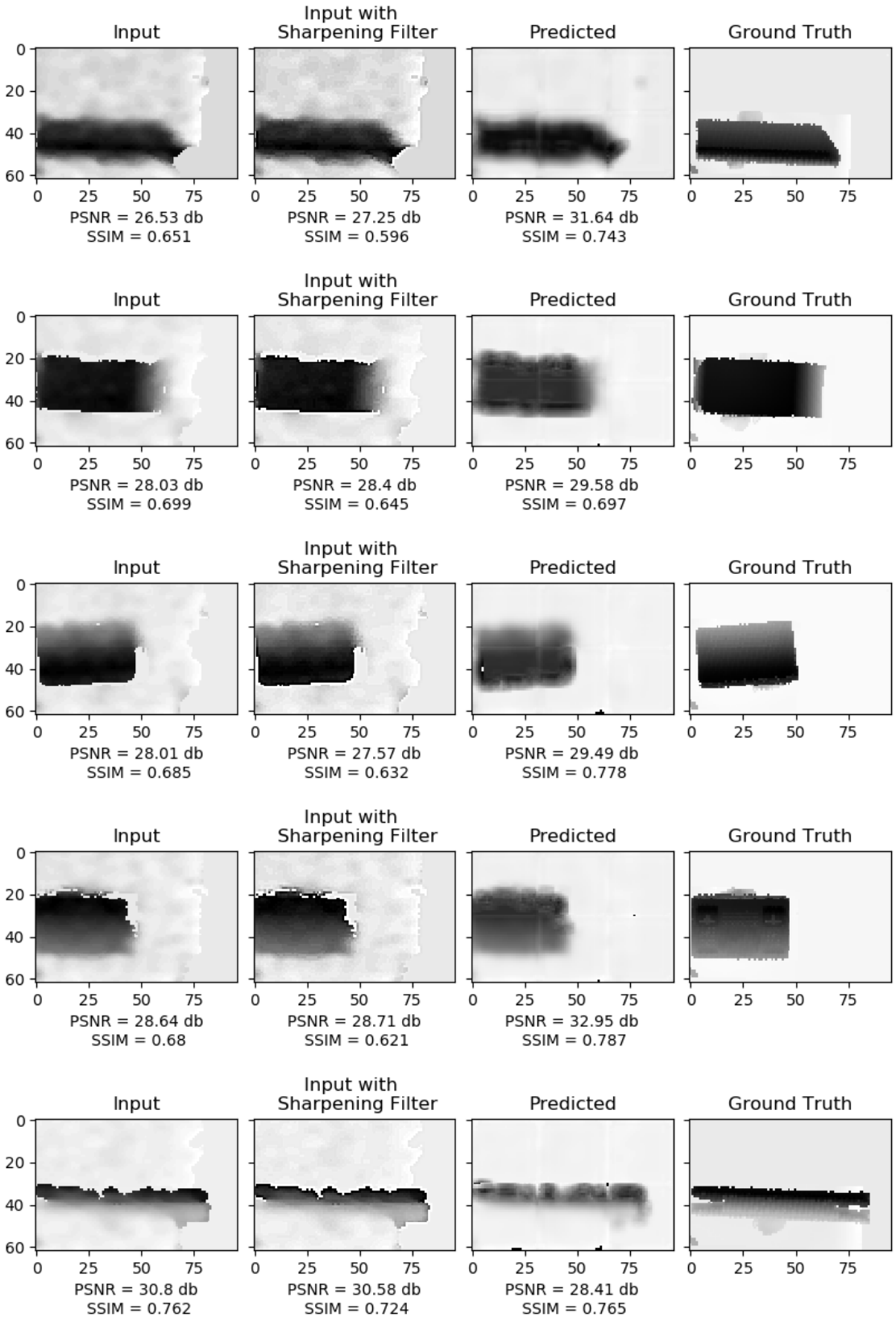
Κατά την εκπαίδευση:

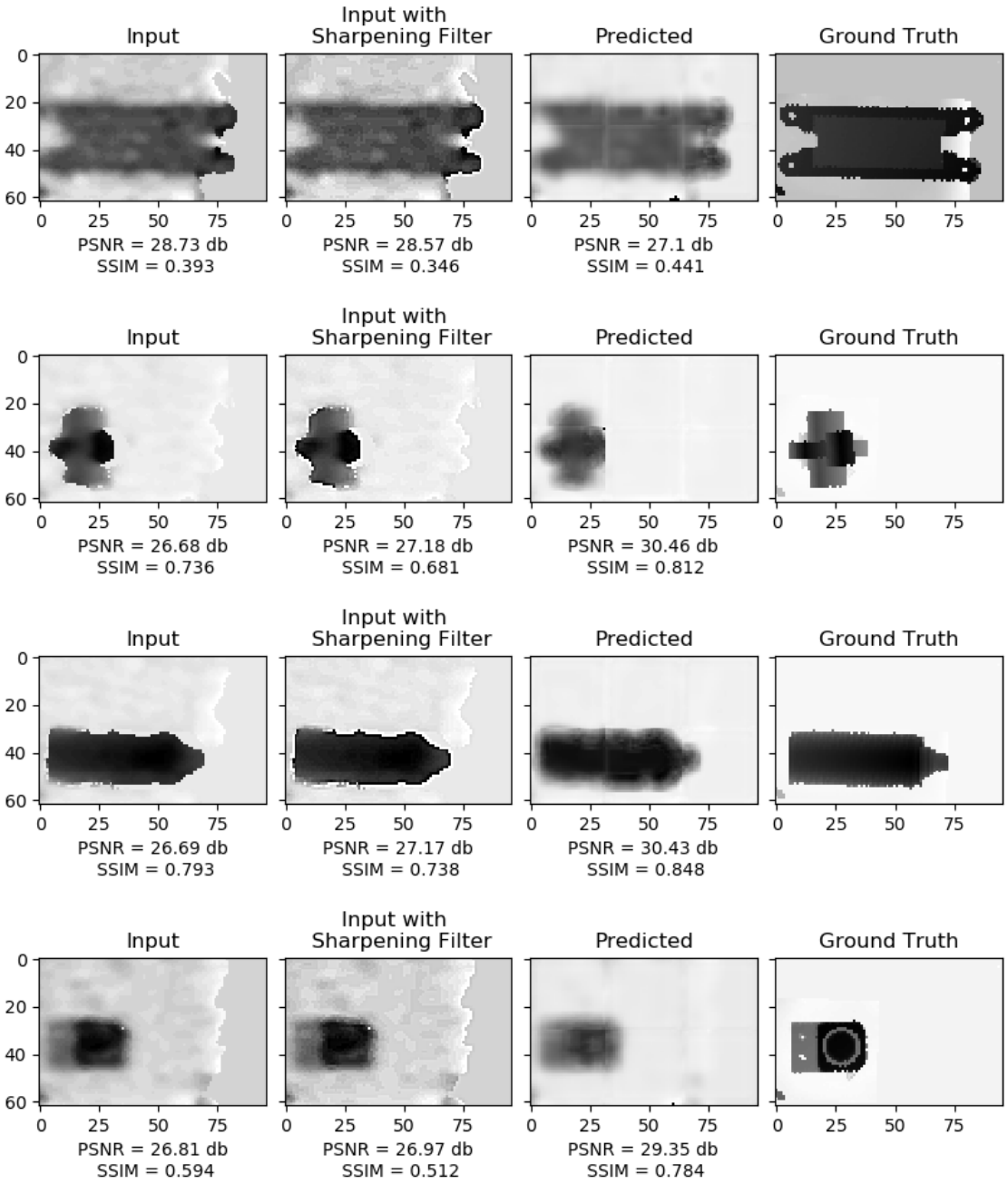
- Εκπαιδεύτηκε για 40 epochs με χρόνο εκπαίδευσης 20 λεπτά, και 30 δευτερόλεπτα
- Ελάχιστο Mean Square Error στο validation set: 0.01879
- Μέγιστο PSNR στο validation set: 17.998 db

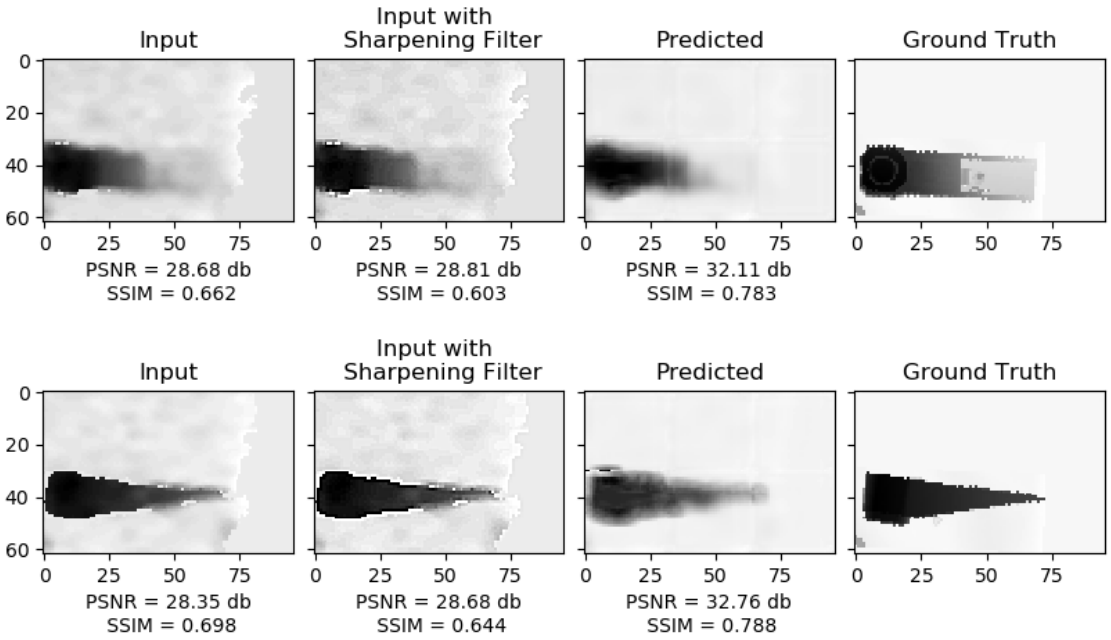
Στο σετ ελέγχου επίδοσης: MSE 0.0162, PSNR 17.883 db

Στις παρακάτω εικόνες παρουσιάζονται και τα 16 αποτελέσματα των επανα-κατασκευασμένων εικόνων βάση του **Error! Reference source not found.** για σύγκριση και με τα αποτελέσματα της αρχιτεκτονικής 128-64-32-1.









Συμπεράσματα

Η εργασία βασίστηκε στο SRCNN όπου πέτυχε αρκετά ικανοποιητικά αποτελέσματα με μικρούς χρόνους εκπαίδευσης. Δοκιμάστηκαν όλα τα πειράματα που αναφέρθηκαν και στην αρχική εργασία[3] για να βρεθεί μια ικανοποιητική αρχιτεκτονική που θα πετύχει τον σκοπό της αποθορυβοποίησης των εικόνων βάθους. Βασικές διαφορές εξαρχής ήταν:

- Τα πακέτα δεδομένων που χρησιμοποίησαν διέφεραν κατά πολύ με το dataset εικόνων βάθους που χρησιμοποιήθηκε για αυτήν την διπλωματική και στον θόρυβο των εικόνων εισόδου και στο πλήθος των εικόνων που χρησιμοποιήθηκαν για εκπαίδευση.
- Το PSNR που πέτυχαν ήταν αρκετά υψηλό. Σε αυτό οφείλεται και ο ακανόνιστος θόρυβος που παρουσίασαν οι εικόνες χαμηλής ποιότητας της RealSense σε σχέση με την τεχνική μεγέθυνσης-σμίκρυνσης των εικόνων του SRCNN, αλλά και στο σχετικά μικρό dataset που χρησιμοποιήθηκε.

Συμπεράσματα σύγκρισης με SRCNN

Τα συμπεράσματα που προκύπτουν σε σύγκριση του αντικειμένου της εργασίας και του SRCNN είναι:

1. Για μεγαλύτερο αριθμό φίλτρων τα αποτελέσματα ήταν καλύτερα.
2. Βαθύτερες αρχιτεκτονικές με περισσότερα κρυφά στρώματα, όντως δεν κατάφερναν να φέρουν καλύτερα αποτελέσματα στις εικόνες βάθους, σε σύγκριση με τα τριών στρωμάτων μοντέλα. Ενδεχομένως βαθύτερες αρχιτεκτονικές να τα κατάφερναν καλύτερα σε εκπαίδευση με περισσότερα δεδομένα.
   Στο τελικό σετ δεδομένων η αρχιτεκτονική των 4 στρωμάτων (Αρχιτεκτονική 128-64-32-1), κατάφερε να δώσει καλύτερα αποτελέσματα σε σχέση με τις αρχιτεκτονικές τριών στρωμάτων.
3. Μεγαλύτερο μέγεθος kernel σε αντίθεση με το SRCNN στο dataset 21x32 (b) όπου εξετάστηκαν τέτοιες αρχιτεκτονικές και συγκεκριμένα η σύγκριση του 9-1-5 με το 11-1-5 ως μέγεθος φίλτρων, δεν απέδωσε καλύτερα από το αρχικό το μικρότερο μέγεθος.
4. Με **μεγαλύτερο μέγεθος φίλτρων στο δεύτερο στρώμα** βάση των δοκιμών που έγιναν, **έδωσαν καλύτερο αποτέλεσμα** και συμφωνούν με το συμπέρασμα του SRCNN ότι λαμβάνοντας μεγαλύτερο μέγεθος γειτονικών πληροφοριών pixel στο στάδιο αντιστοίχισης (mapping) όντως οδηγεί σε καλύτερα αποτελέσματα.

Συμπεράσματα πορείας αποτελεσμάτων

Για το πακέτο δεδομένων 42x64 όπως αναφέρθηκε δεν εξάγονται συμπεράσματα και παρουσιάστηκε για την πορεία των αποτελεσμάτων και για την εξοικείωση με το πρόβλημα. Για το πρώτο dataset 21x32 με **11904 εικόνες** τα συμπεράσματα που προκύπτουν είναι:

- Για μικρό αριθμό φίλτρων όπως το 64-32-1 τα αποτελέσματα ήταν ιδανικά σε σχέση με τα μεγαλύτερου αριθμού φίλτρων, καθώς το δίκτυο αν και εκπαιδεύτηκε χωρίς μέθοδο πρόωρου σταματήματος (early stopping) δεν παρουσίασε υπερβολική τοποθέτηση (overfitting) και κατάφερε ένα ικανοποιητικό PSNR με ελαφρώς

βελτιωμένη εικόνα εξόδου, σε σχέση με την εικόνα εισόδου, ως προς το περίγραμμα και το βάθος της εικόνας. Αν και τα αποτελέσματα περιείχαν θόρυβο με τις κατάλληλες βελτιώσεις θα μπορούσε να χρησιμοποιηθεί σε μια εφαρμογή.

- Στα μοντέλα με μεγαλύτερο αριθμό φίλτρων παρουσιάστηκε overfitting στον τρόπο με τον οποίο εκπαιδεύτηκε το δίκτυο . Overfitting που έφερε αρνητικά αποτελέσματα στο σετ ελέγχου επίδοσης. Ο λόγος που δεν εξετάστηκε εκ νέου οι συγκεκριμένες αρχιτεκτονικές με μέθοδο «early stopping» είναι γιατί και το σετ εκπαίδευσης είναι μικρό, αλλά και τα σετ ελέγχου επίδοσης και επικύρωσης με 2 εικόνες δεν θα είναι αντιπροσωπευτικό για αξιόπιστα αποτελέσματα σε ένα δίκτυο CNN.
- Βάση των παραπάνω, συμπεραίνεται ότι για το συγκεκριμένο dataset, η αρχιτεκτονική (64-32-1, 9-1-5), που δανείστηκε για εξέταση από το SRCNN, πέτυχε ικανοποιητικά αποτελέσματα αποθορυβοποίησης εικόνων βάθους, αν και το μέγεθος του πακέτου δεδομένων δεν είναι ιδανικό για ισχυρά συμπεράσματα και προτάσεις. Αυτός είναι και ο λόγος που επιλέχθηκε επαύξηση δεδομένων για να δημιουργηθεί ένα μεγαλύτερο σετ στα επόμενα πειράματα. Επίσης και το μέγεθος των δεδομένων ελέγχου επίδοσης (test), είναι μικρό για αξιόλογα συμπεράσματα.
- Προτάσεις βελτίωσης δεν δίνονται για το συγκεκριμένο dataset.

Για το δεύτερο dataset με τις **23936 εικόνες**, σχεδόν διπλάσιες του προηγούμενου τα συμπεράσματα που προκύπτουν είναι:

- Το δίκτυο είχε πολύ μεγαλύτερο σετ επικύρωσης για να εξάγει καλύτερα βάρη από την εκπαίδευση μέσω του «early stopping».
- Το δίκτυο κατάφερε να μειώσει τον θόρυβο περιβάλλοντος (γενικό θόρυβο πέρα των αντικειμένων) βάση των αποτελεσμάτων που παρουσιάστηκαν, στις εικόνες LR. Αυτό είναι θετικό και θα μπορούσε να χρησιμοποιηθεί σε εφαρμογές που απαιτούν καλύτερα αποτελέσματα χωρίς θόρυβο.
- Τα αποτελέσματα στο βάθος και στο περίγραμμα ήταν ανάμεικτα, καθώς λεπτά περιγράμματα όπως η άκρη του κατσαβιδιού που παρουσιάστηκε στα αποτελέσματα, αντιμετωπίστηκε ως θόρυβος από το δίκτυο και έτεινε να την εξαλείψει. Σε μερικές από τις εικόνες του σετ ελέγχου επίδοσης βελτίωσε και το βάθος και το περίγραμμα και παρουσίασε θετικά αποτελέσματα. Σε κάθε αρχιτεκτονική στα αποτελέσματα αναλύεται περαιτέρω η επίδοση του δικτύου.
- Καλύτερες αρχιτεκτονικές που προέκυψαν είναι οι (64-32-1, 9-1-5) και η (256-128-1,9-1-5) με τα καλύτερα αποτελέσματα φαινομενικά να είναι της πρώτης εξ' αυτών, ενώ σε σύγκριση και με τον χρόνο εκπαίδευσης τότε καλύτερη αρχιτεκτονική προς εφαρμογή είναι καθαρά η (64-32-1, 9-1-5).
- Βάση των παραπάνω συμπεραίνεται ότι τα αποτελέσματα δεν είναι ικανοποιητικά για υπερ-ανάλυση εικόνων βάθους καθώς όπως φάνηκε και στον σχολιασμό των αποτελεσμάτων το 50% των εικόνων του σετ ελέγχου επίδοσης βελτιώθηκαν και σε βάθος και στο περίγραμμα ενώ το υπόλοιπο 50% ίσως να χειροτέρεψε. Πρακτικά αυτό σημαίνει ότι δεν θα τα κατάφερνε καλά σε μεγάλο ποσοστό εικόνων βάθους. Σε αυτό το ποσοστό συνέβαλε και το μικρό σετ ελέγχου επίδοσης που χρησιμοποιήθηκε.

- Τα μη αξιόλογα αποτελέσματα οδήγησαν σε ανάγκη για βελτιστοποίηση και περαιτέρω πειραματισμό. Γι' αυτό, μιας και δεν υπήρχε καινούριο ανεπεξέργαστο πακέτο δεδομένων, χρησιμοποιήθηκε περισσότερες τεχνικές για την δημιουργία περισσότερων εικόνων (data augmentation). Το μεγαλύτερο πακέτο δεδομένων θεωρητικά θα ανέβαζε το ποσοστό επιτυχίας αποθορυβοποίησης εικόνων βάθους.

Η πορεία των μέχρι τώρα πειραμάτων όπως επίσης και των εξόδων των προηγούμενων δικτύων που δεν κατάφεραν να βελτιώσουν σημαντικά τις εικόνες βάθους οδήγησε στην δημιουργία ακόμα μεγαλύτερου dataset του 32x32 με **36972 εικόνες**. Σε αυτό μπορεί να οφείλεται και τυχόν λάθη που προέκυψαν στη μεθοδολογία επιλογής dataset. Τα συμπεράσματα που προκύπτουν από το τελικό πακέτο δεδομένων είναι:

- Αν και δοκιμάστηκε για τις διαστάσεις 21x32 ένα dataset 89000 που περιείχε τα ίδια βήματα για την επαύξηση δεδομένων με το 32x32, με πιο μικρό stride τα αποτελέσματα δεν ήταν ικανοποιητικά, σε σχέση με το μεγαλύτερο stride του 32x32. Που συμπεραίνεται ότι η **μεθοδολογία με πολύ μικρό stride (3pixel) ήταν λανθασμένη**. Επίσης, ένα σετ δεδομένων ελέγχου επίδοσης όπως αυτό που κρατήθηκε στις τελευταίες δοκιμές, που ανέρχεται σε 16 εικόνες και είναι περίπου στο 17% του αρχικού dataset προσέφερε πιο αξιόπιστα συμπεράσματα για την επίδοση του δικτύου.
- Τα αποτελέσματα συγκριτικά, βάση του πίνακα Πίνακας 2-2 δείχνουν ότι αρκετές αρχιτεκτονικές στο σετ ελέγχου επίδοσης πέτυχαν υψηλότερο PSNR από τις εικόνες βάθους της RealSense με ποσοστό βελτίωσης 70%. Σε συνδυασμό και με το υψηλό SSIM που δηλώνει την ομοιότητα μεταξύ εικόνων με την βασική εικόνα (ground truth).
- Υψηλότερο PSNR και SSIM και από τεχνική sharpening. Το φίλτρο sharpening αν και στο μεγαλύτερο ποσοστό εικόνων αύξησε το PSNR που δηλώνει το απόλυτο σφάλμα pixel, και κατάφερε να βελτιώσει το περίγραμμα του αντικειμένου απεικόνισης. Δεν κατάφερε και αντίθετα χειροτέρεψε τον γενικό θόρυβο στις εικόνες βάθους.
- Τα μοντέλο που έδωσε τα καλύτερα αποτελέσματα και προτείνεται μέσα από την εργασία είναι η **Αρχιτεκτονική 256-128-1**. Η συγκεκριμένη αρχιτεκτονική χρειάστηκε αρκετό χρόνο εκπαίδευσης αλλά τα αποτελέσματα που παρουσίασε μπορούν να δώσουν χρήσιμα συμπεράσματα όπως:
    - Κατάφερε να **εξαλείψει τον γενικό θόρυβο**, που πρακτικά σημαίνει βάθος εκτός και μακριά του αντικειμένου, που πρόσθεσε στα δεδομένα της η εικόνα βάθους της RealSense. Το αποτέλεσμα αυτό είναι αρκετά σημαντικό για χρήση της συγκεκριμένης κάμερας σε εφαρμογές που τυχόν βάθος σε ολόκληρο το οπτικό πεδίο θα δώσει εσφαλμένες μετρήσεις.
    - Στο απόλυτο βάθος του αντικειμένου απεικόνισης (διαβάθμιση γκρι), καθώς επίσης και το περίγραμμα, τα αποτελέσματα όπως φαίνονται και στις εικόνες που επισυνάφτηκαν στο κεφάλαιο της αρχιτεκτονικής, πολλές φορές δεν το πρόβλεψαν σε μεγάλο βαθμό. Βάση αυτού συνεπάγεται ότι σε **εφαρμογές που είναι πιο σημαντικό το περίγραμμα και η απόσταση**

του αντικειμένου από τον αισθητήρα δεν προτείνεται η χρήση των αλγόριθμων της εργασίας για αποθορυβοποίηση.

Βάση όλων των παραπάνω και της πορείας της εργασίας τα συνοπτικά συμπεράσματα καταλήγουνε σε:

1. Η μεθοδολογία του 3^{ου} dataset με διαστάσεις 32x32 και η αρχιτεκτονική που έδωσε τα καλύτερα αποτελέσματα, δύναται να απαντήσει στο ***αρχικό ερευνητικό ερώτημα*** αν μπορεί να βελτιώσει τα δεδομένα της Realsense. Έτσι προκύπτει ότι η χρήση των αλγορίθμων CNN που παρουσιάστηκαν μέσα από την εργασία, **προτείνεται για την χρήση σε εφαρμογές όπου η αποθορυβοποίηση στο σύνολο του οπτικού πεδίου της κάμερας βάθους RealSense είναι αρκετά σημαντική, σε συνδυασμό με μικρή βελτίωση στο περίγραμμα και στο βάθος του αντικειμένου.**
2. Το μεγαλύτερο dataset δίνει καλύτερα αποτελέσματα στην εποπτευόμενη μάθηση και με αρχιτεκτονικές με μικρότερο βάθος όπως αυτές του SRCNN μπορεί να βελτιώσει τα αποτελέσματα σε σχέση με απλά φίλτρα.

Προτάσεις βελτιστοποίησης και περεταίρω πειραματισμού

Στα CNN όπως και σε όλες τις εφαρμογές της μηχανικής μάθησης βάση της βιβλιογραφίας, όσο το δυνατόν περισσότερα δεδομένα χρησιμοποιηθούν για εκπαίδευση, τόσο καλύτερα αποτελέσματα θα φέρουν στην τελική έξοδο. Αυτό ισχύει για τεχνικές αποθορυβοποίησης ή υπερ-ανάλυσης όπως και σε άλλα προβλήματα π.χ πρόγνωσης (prediction), κατηγοριοποίησης (classification) κ.α.

Για το υπάρχον πρόβλημα που εξετάστηκε η μη κανονικότητα του θορύβου που έδωσε η stereo depth κάμερα της real sense, ήταν αρκετά απρόβλεπτη για το CNN. Ειδικά μέσα από ένα μικρό dataset 95 εικόνων δεν ήταν αρκετό για να μπορέσει να αναγνωρίσει σε ορισμένες περιπτώσεις, κατάλληλα τα πρότυπα. Βάση αυτού κύρια πρόταση βελτιστοποίησης είναι:

1. Μεγαλύτερο ανεπεξέργαστο dataset που θα βοηθήσει σε καλύτερη αντιστοιχία εικόνων βάθους του CNN.
2. Επιπλέον, μπορεί να χρησιμοποιηθεί μεγαλύτερο dataset από το ήδη υπάρχον, δημιουργώντας περισσότερες εικόνες βάθους από τα νέφη σημείων με διαφορετική γωνία θέασης. Καθώς για την συγκεκριμένη εφαρμογή χρησιμοποιήθηκε η γωνία θέασης 90^{o} των νεφών, που μετατράπηκαν σε εικόνες βάθους. Αυτό πρακτικά σημαίνει ότι, για τα ίδια δεδομένα, μπορούμε να δημιουργήσουμε ένα πολύ μεγαλύτερο dataset για αντικείμενα μη κανονικού κατανεμημένου όγκου και σχήματος. Αντικείμενα δηλαδή, που το σχήμα τους δεν εμπίπτει σε στερεά όπως σφαίρα, ορθογώνιο παραλληλόγραμμο κ.α. που η διαφορετική γωνία θέασης, θα δημιουργήσει διαφορετικές εικόνες του αντικειμένου. Με αυτή τη μέθοδο ουσιαστικά, αξιοποιούνται τα οφέλη της τρισδιάστατης απεικόνισης.

Η συνάρτηση κόστους μέσων τετραγώνων (Mean Square Error), συγκρίνει τις τιμές pixel της εικόνας εξόδου και της εικόνας Ground Truth. Σε άρθρα που αναφέρονται σε προβλήματα υπερ-ανάλυσης και αποθορυβοποίησης [32], προτείνουν διαφορετικές συναρτήσεις

κόστους που μπορούν να βελτιώσουν τα τελικά αποτελέσματα με σημείο αναφοράς ολόκληρη την ποιότητα εικόνας. Σαν δεύτερη πρόταση βελτιστοποίησης λοιπόν, προτείνεται:

3. Πειραματισμός με διαφορετική συνάρτηση κόστους. Σε συνδυασμό με ένα μεγαλύτερο ανεπεξέργαστο dataset από αυτό που εξετάστηκε και με βάση τα αποτελέσματα που αναφέρθηκαν, μπορεί να ερευνηθεί το αποτέλεσμα με διαφορετικές συναρτήσεις κόστους που θα υπολογίζουν την γενική ποιότητα εικόνας , όπως [32]:
    a. Κόστος περιεχομένου (Content Loss). Για παράδειγμα μπορούμε να κρατήσουμε τα συνελικτικά στρώματα εκπαιδευμένων (State of the Art) μοντέλων όπως του VGG19[43] και να τα χρησιμοποιήσουμε σαν συνάρτηση κόστους. Στην εργασία στην προσπάθεια υλοποίησης μιας τέτοιας αρχιτεκτονικής , το hardware του Η/Υ δεν κατάφερε να ανταπεξέλθει στην ανάγκη για RAM μιας τέτοιας αρχιτεκτονικής και τερματίστηκε ανεπιτυχώς.
    b. Κόστος «υφής» (Texture Loss)
    c. Συνολικό κόστος μεταβολής (Total Variation Loss)

Επίσης διαφορετικές αρχιτεκτονικές και τεχνικές που θα χρησιμοποιήσουν είτε εποπτευόμενη, είτε μη εποπτευόμενη μάθηση, μπορούν να βελτιώσουν κατά πολύ τα αποτελέσματα και σε συνδυασμό με τις προηγούμενες προτάσεις να καταλήξουν σε μια λύση (State of The Art) πάνω στο συγκεκριμένο πρόβλημα της αποθορυβοποίησης και υπερ-ανάλυσης εικόνων βάθους. Βάση αυτών η τρίτη και τελευταία πρόταση βελτιστοποίησης που προτείνεται από αυτή την εργασία είναι:

4. Περαιτέρω πειραματισμός με αρχιτεκτονικές και τεχνικές CNN. Μερικές από αυτές που βάση της μέχρι τώρα έρευνας έδωσαν αρκετά ικανοποιητικά αποτελέσματα είναι:
    a. Εκτός από τις κλασσικά συνελικτικά στρώματα, παραλλαγές στην αρχιτεκτονική μπορούν να χρησιμοποιηθούν για καλύτερα αποτελέσματα όπως του MWCNN [7], residual learning, batch normalization, skip connections στα στρώματα κ.α.[4]
    b. Αρχιτεκτονικές GANs (Generative Adversarial Networks)[13] όπου έχουν πετύχει εκπληκτικά αποτελέσματα σε προβλήματα μηχανικής όρασης. Λειτουργούν σαν δύο νευρωνικά δίκτυα, όπου το ένα δημιουργεί καινούρια δεδομένα βάση των δεδομένων εκπαίδευσης, ενώ το άλλο δίκτυο κατηγοριοποιεί τις εικόνες που δημιουργήθηκαν αν είναι ρεαλιστικές ή όχι. Τα δύο μοντέλα εκπαιδεύονται ταυτόχρονα μέχρι το μοντέλο ελέγχου (discriminator) ξεγελαστεί από το μοντέλο δημιουργίας δεδομένων (generator).

Επιπλέον χρήζει εξέτασης, αν στις βελτιωμένες (αποθορυβοποιημένες) εικόνες εξόδου των CNN, η εφαρμογή παραδοσιακών τεχνικών επεξεργασίας εικόνων μέσω φίλτρων (όπως αυτά που αναφέρθηκαν στην εισαγωγή), μπορεί να συνδυαστεί για ακόμα καλύτερα αποτελέσματα.

Κατάλογος Εικόνων και Πινάκων

Πηγές

Ελληνική Βιβλιογραφία

1. Ι. Βλαχάβας, Π. Κεφαλάς, Ν. Βασιλειάδης, Φ. Κόκκορας, Η. Σακελλαρίου, "Τεχνητή Νοημοσύνη - Γ' Έκδοση", Εκδόσεις Πανεπιστημίου Μακεδονίας, Θεσσαλονίκη, 2011
2. Γεωργούλη, Α. 2015. Τεχνητή νοημοσύνη. [ηλεκτρ. βιβλ.] Αθήνα: Σύνδεσμος Ελληνικών Ακαδημαϊκών Βιβλιοθηκών. κεφ 4. Διαθέσιμο στο: http://hdl.handle.net/11419/3382

Ξενόγλωσση Βιβλιογραφία

3. C. Dong, C. C. Loy, K. He and X. Tang, "Image Super-Resolution Using Deep Convolutional Networks," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 38, no. 2, pp. 295-307, 1 Feb. 2016, doi: 10.1109/TPAMI.2015.2439281.
4. Z. Wang, J. Chen and S. C. H. Hoi, "Deep Learning for Image Super-resolution: A Survey," in IEEE Transactions on Pattern Analysis and Machine Intelligence, doi: 10.1109/TPAMI.2020.2982166.
5. Ian Goodfellow, Yoshua Bengio and Aaron Courville, "Deep Learning" MIT Press https://www.deeplearningbook.org/contents/intro.html
6. N. Dimitriou et al., "Fault Diagnosis in Microelectronics Attachment Via Deep Learning Analysis of 3-D Laser Scans," in IEEE Transactions on Industrial Electronics, vol. 67, no. 7, pp. 5748-5757, July 2020, doi: 10.1109/TIE.2019.2931220.
7. P. Liu, H. Zhang, K. Zhang, L. Lin and W. Zuo, "Multi-level Wavelet-CNN for Image Restoration," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Salt Lake City, UT, USA, 2018, pp. 886-88609, doi: 10.1109/CVPRW.2018.00121.
8. Sandra Vieira, Walter Hugo Lopez Pinaya, Andrea Mechelli "Using deep learning to investigate the neuroimaging correlates of psychiatric and neurological disorders: Methods and applications" King's College London
9. Normand J. Beaudry, Renato Renner "AN INTUITIVE PROOF OF THE DATA PROCESSING INEQUALITY", Institute for Theoretical Physics, ETH Zurich, Wolfgang-Pauli-Str. 27 8093 Zurich, Switzerland
10. Zhou Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," in IEEE Transactions on Image Processing, vol. 13, no. 4, pp. 600-612, April 2004, doi: 10.1109/TIP.2003.819861.
11. Olga Russakovsky*, Jia Deng*, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg and Li Fei-Fei. (* = equal contribution) ImageNet Large Scale Visual Recognition Challenge. IJCV, 2015. paper | bibtex | paper content on arxiv
12. Kingma, Diederik & Ba, Jimmy. (2014). Adam: A Method for Stochastic Optimization. International Conference on Learning Representations.
13. C. Ledig *et al.*, "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2017, pp. 105-114, doi: 10.1109/CVPR.2017.19.

14. K. Zeng, J. Yu, R. Wang, C. Li and D. Tao, "Coupled Deep Autoencoder for Single Image Super-Resolution," in IEEE Transactions on Cybernetics, vol. 47, no. 1, pp. 27-37, Jan. 2017, doi: 10.1109/TCYB.2015.2501373.
15. A. Horé and D. Ziou, "Image Quality Metrics: PSNR vs. SSIM," 2010 20th International Conference on Pattern Recognition, Istanbul, Turkey, 2010, pp. 2366-2369, doi: 10.1109/ICPR.2010.579.
16. Z. -S. Liu, W. -C. Siu and Y. -L. Chan, "Reference Based Face Super-Resolution," in IEEE Access, vol. 7, pp. 129112-129126, 2019, doi: 10.1109/ACCESS.2019.2934078.

Ιστοσελίδες

17. https://el.wikipedia.org/wiki/%CE%A4%CE%B5%CF%87%CE%BD%CE%B7%CF%84%CE%AE_%CE%BD%CE%BF%CE%B7%CE%BC%CE%BF%CF%83%CF%8D%CE%BD%CE%B7
18. https://el.wikipedia.org/wiki/Python
19. https://el.wikipedia.org/wiki/%CE%9C%CE%B7%CF%87%CE%B1%CE%BD%CE%B9%CE%BA%CE%AE_%CF%8C%CF%81%CE%B1%CF%83%CE%B7
20. https://el.wikipedia.org/wiki/%CE%8C%CF%81%CE%B1%CF%83%CE%B7
21. https://el.wikipedia.org/wiki/%CE%97%CE%BB%CE%B5%CE%BA%CF%84%CF%81%CE%BF%CE%BC%CE%B1%CE%B3%CE%BD%CE%B7%CF%84%CE%B9%CE%BA%CE%AE_%CE%B1%CE%BA%CF%84%CE%B9%CE%BD%CE%BF%CE%B2%CE%BF%CE%BB%CE%AF%CE%B1
22. https://el.wikipedia.org/wiki/%CE%97%CE%BB%CE%B5%CE%BA%CF%84%CF%81%CE%BF%CE%BC%CE%B1%CE%B3%CE%BD%CE%B7%CF%84%CE%B9%CE%BA%CF%8C_%CF%86%CE%AC%CF%83%CE%BC%CE%B1
23. https://en.wikipedia.org/wiki/Lidar
24. https://www.scratchapixel.com/index.php
25. https://www.intelrealsense.com/beginners-guide-to-depth/
26. https://el.wikipedia.org/wiki/%CE%9C%CE%B7%CF%87%CE%B1%CE%BD%CE%B9%CE%BA%CE%AE_%CE%BC%CE%AC%CE%B8%CE%B7%CF%83%CE%B7
27. https://en.wikipedia.org/wiki/Convolutional_neural_network#Convolutional
28. http://www.open3d.org/docs/latest/tutorial/Advanced/pointcloud_outlier_removal.html
29. https://github.com/titu1994/Image-Super-Resolution/tree/dd5149b4632f35a4148a97c3aa77c6fa79d9d2d9
30. https://www.optimet.com/conopoint-10.php
31. https://machinelearningmastery.com/adam-optimization-algorithm-for-deep-learning/
32. https://beyondminds.ai/blog/an-introduction-to-super-resolution-using-deep-learning/
33. https://github.com/tensorflow/tensorflow/blob/v2.4.1/tensorflow/python/ops/image_ops_impl.py#L3885
34. https://www.cns.nyu.edu/~lcv/ssim/
35. https://en.wikipedia.org/wiki/Noise_reduction#In_images
36. https://sinestesia.co/blog/tutorials/python-cube-matrices/
37. https://en.wikipedia.org/wiki/Point_set_registration#Algorithms
38. https://www.intelrealsense.com/depth-camera-d435/

39. Peak signal-to-noise ratio, https://en.wikipedia.org/w/index.php?title=Peak_signal-to-noise_ratio&oldid=1001592263
40. https://medium.com/analytics-vidhya/creating-an-autoencoder-with-pytorch-a2b7e3851c2c
41. https://el.wikipedia.org/wiki/%CE%94%CE%B9%CE%B1%CE%BA%CF%8D%CE%BC%CE%B1%CE%BD%CF%83%CE%B7
42. https://theaisummer.com/skip-connections/
43. https://arxiv.org/abs/1409.1556
44. https://keras.io/api/applications/vgg/