ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

Σχολή Αγρονόμων και Τοπογράφων Μηχανικών - Μηχανικών Γεωπληροφορικής

Εργαστήριο Τηλεπισκόπησης

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΑΞΙΟΠΟΙΗΣΗ ΤΟΥ ΜΗΧΑΝΙΣΜΟΥ ΠΡΟΣΟΧΗΣ ΓΙΑ ΑΠΟΔΟΤΙΚΟΤΕΡΗ ΧΡΗΣΗ ΧΑΡΑΚΤΗΡΙΣΤΙΚΩΝ ΣΕ ΑΥΤΟ-ΕΠΙΒΛΕΠΟΜΕΝΑ ΜΟΝΤΕΛΑ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ

Ειρήνη Μπαλτζή

Επιβλέπων: Καθ. Κ. Καράντζαλος

ΑΘΗΝΑ, ΙΟΥΛΙΟΣ 2024

NATIONAL TECHNICAL UNIVERSITY OF ATHENS

School of Rural, Surveying and Geoinformatics Engineering

Remote Sensing Lab

DIPLOMA THESIS

ATTENTIVE PROBING: LEVERAGING ATTENTION FOR IMPROVED FEATURE UTILIZATION IN SELF-SUPERVISED MODELS

Eirini Baltzi

Supervisor: Prof. K. Karantzalos

ATHENS, JULY 2024

RSLab

Remote Sensing Laboratory

National Technical University of Athens

✓ Sensing ✓ Analytics ✓ Monitoring

Περίληψη

Η παρούσα διπλωματική εργασία προτείνει μια νέα, μη γραμμική μέθοδο χρήσης χαρακτηριστικών, χρησιμοποιώντας τον μηχανισμό προσοχής, ως εναλλακτική της παραδοσιακής γραμμικής χρήσης. Η προτεινόμενη μέθοδος, βασισμένη στη μέθοδο συγκέντρωσης SimPool [20], ονομάζεται *προσεκτική χρήση*, καθώς εισάγει τον μηχανισμό διασταυρούμενης προσοχής στη διαδικασία, βελτιώνοντας έτσι τη γενική αναπαράσταση των χαρακτηριστικών. Για την αξιολόγηση της μεθόδου, πραγματοποιούνται πειράματα χρήσης χαρακτηριστικών για την ταξινόμηση εικόνων χρησιμοποιώντας τρία σύγχρονα, προ-εκπαιδευμένα, αυτο-επιβλεπόμενα μοντέλα: MAE [15], DINO [11] και DINOv2 [19]. Τα πειρματικά αποτελέσματα δείχνουν ότι η προτεινόμενη μέθοδος υπερβαίνει σε απόδοση τόσο τις απλές, όσο και άλλες πιο περίπλοκες μεθόδους, προσφέροντας υψηλή ακρίβεια ταξινόμησης και παράλληλα χαμηλό υπολογιστικό κόστος. Ακόμα, η μέθοδος χαρακτηρίζεται από την βελτιωμένη ερμηνευσιμότητα της σε σχέση με άλλες μεθόδους, καθώς παράγει χάρτες προσοχής υψηλής ακρίβειας που οριοθετούν τα όρια των αντικειμένων των εικόνων. Επιτρέπει έτσι την βαθύτερη κατανόηση των αλληλεπιδράσεων και της σημασίας των χαρακτηριστικών, καθιστώντας την ιδιαίτερα χρήσιμη για την αξιολόγηση αυτο-επιβλεπόμενων μοντέλων. Συμπεριλαμβανομένων των υψηλών ακριβειών που προσφέρει και του χαμηλού υπολογιστικό κόστους που απαιτεί, η μέθοδος μπορεί να χρησιμοποιηθεί και σε άλλες κατάντη εργασίες, όπως η σημασιολογική τμηματοποίηση εικόνων και η ανίχνευση αντικειμένων, προσφέροντας μια ευρύτερη εφαρμογή και χρησιμότητα στο πεδίο των εφαρμογών της Όρασης Υπολογιστών.

Λέξεις-Κλειδιά: Γραμμική Χρήση Χαρακτηριστικών, Προσεκτική Χρήση Χαρακτηριστικών, Μηχανισμός Προσοχής, Ταξινόμηση Εικόνας, Αυτο-επιβλεπόμενη Μάθηση, Νευρωνικά Δίκτυα

Abstract

In this study, we propose a novel, non-linear probing method, utilizing an attention mechanism, as an alternative to traditional linear probing. The proposed method, based on SimPool [20], is essentially an *attentive probing*, as it introduces the cross-attention mechanism into the process, thereby improving the global image representation. To evaluate the method, probing experiments are conducted for image classification employing three modern, pre-trained, self-supervised models: MAE [15], DINO [11], and DINOv2 [19]. The experimental results show that our attentive probing outperforms both traditional linear probing, as well as other attentive pooling methods repurposed as attentive probing, offering high classification accuracy while maintaining low computational cost. Additionally, it is characterized by its improved interpretability compared to other methods, as it produces high-quality attention maps that delineate the boundaries of image objects. This allows for a deeper understanding of the interactions and significance of features, making it particularly useful for the evaluation of self-supervised models. Given its high accuracy and low computational requirements, the method can also be used in other downstream tasks, such as semantic segmentation and object detection, offering broader applicability and utility in the field of Computer Vision.

Keywords: Attentive Probing, Linear Probing, Attention Mechanism, Image Classification, Self-supervised Learning, Neural Networks

RSLab
Remote Sensing Laboratory
National Technical University of Athens
√ Sensing √ Analytics √ Monitoring

Ευχαριστίες

Με την εκπόνηση της παρούσας διπλωματικής εργασίας θα ήθελα να ευχαριστήσω θερμά τον επιβλέπων καθηγητή Κωνσταντίνο Καράντζαλο για την εμπιστοσύνη, την υποστήριξη και τις συμβουλές που μου έδωσε καθόλη τη διάρκεια της συνεργασίας μας. Θα ήθελα επίσης να ευχαριστήσω τον Υπ. Διδάκτορα Βασίλη Ψωμά, για την καθοδήγηση και βοήθεια όλα αυτά τα χρόνια. Οι συμβουλές του, η συνεχής ενασχόληση και η άψογη συνεργασία μας ήταν καθοριστικά για την εκπόνηση της παρούσας εργασίας. Θα ήθελα ακόμα να ευχαριστήσω τον Διονύση Χριστόπουλο και τον Ιωάννη Κακογεωργίου για την πολύ καλή ομάδα και συνεργασία που έχουμε και συνεχίζουμε ώστε η διπλωματική να προβεί και σε δημοσίευση. Τέλος, θα ήθελα να ευχαριστήσω την οικογένεια μου και τους φίλους μου για την στήριξη τους καθόλη την διάρκεια των σπουδών μου.

Ειρήνη Μπαλτζή
Αθήνα, Ιούλιος 2024

Περιεχόμενα

Κατάλογος Πινάκων

Κατάλογος Σχημάτων

ΚΕΦΑΛΑΙΟ 1

Εισαγωγή

1.1 Κίνητρο

Στην αυτο-επιβλεπόμενη μάθηση (self-supervised learning), η εξαγωγή και αξιοποίηση ισχυρών αναπαραστάσεων (representations) είναι κρίσιμη για την επιτυχία των κατάντη εργασιών (downstream tasks). Παραδοσιακές μέθοδοι, όπως η ακριβής ολική ή μερική ρύθμιση (fine-tuning) του δικτύου, απαιτούν σημαντικά μεγέθη υπολογιστικών πόρων και χρόνου, καθώς ενημερώνουν όλες ή μερικές παραμέτρους του μοντέλου αντίστοιχα. Αυτή η προσέγγιση υποβιβάζει την φάση της αυτο-επιβλεπόμενης μάθησης σε ένα απλό βήμα αρχικοποίησης, καθώς το δίκτυο ουσιαστικά επανεκπαιδεύεται για την κατάντη εργασία. Απλούστερες μέθοδοι, όπως η ταξινόμηση εγγύτερου γείτονα (k-NN classification) και η γραμμική χρήση χαρακτηριστικών (linear probing) ενώ χαρακτηρίζονται υπολογιστικά αποδοτικές, δεν εκμεταλλεύονται πλήρως τα πλούσια χαρακτηριστικά που μαθαίνονται στην προ-εκπαίδευση του μοντέλου με αυτο-επιβλεπόμενη μάθηση. Η ανάγκη για μια νέα προσέγγιση που θα εξισορροπεί τα προαναφερθέντα προβλήματα, είναι μεγάλη. Στην παρούσα διπλωματική εργασία, προτείνεται και ερευνάται μια νέα, μη γραμμική μέθοδος χρήσης χαρακτηριστικών με χρήση του μηχανισμού προσοχής (*attentive probing*), ως εναλλακτική της παραδοσιακής γραμμικής χρήσης. Η μέθοδος στοχεύει στην πλήρη αξιοποίηση των χαρακτηριστικών που έχουν μαθευτεί από τα προ-εκπαιδευμένα μοντέλα αυτο-επιβλεπόμενης μάθησης, διατηρώντας ταυτόχρονα τα πλεονεκτήματα της αποδοτικότητας και της εύκολης εφαρμογής τους.

1.2 Στόχοι

Οι στόχοι της παρούσας εργασίας ποικίλουν και σκοπεύουν στην ανάπτυξη μιας μεθόδου η οποία θα διέπεται από:

1. **Αποδοτικότητα:** Θα διατηρεί όσο το δυνατόν περισσότερο την υπολογιστική αποδοτικότητα της γραμμικής χρήσης χαρακτηριστικών.
2. **Αξιοποίηση Χαρακτηριστικών:** Θα εκμεταλλεύεται πλήρως τα μαθημένα χαρακτηριστικά του προ-εκπαιδευμένου μοντέλου μεσω της χρήσης του μηχανισμού προσοχής, διασφαλίζοντας ότι η γενική αναπαράσταση (global representation) λαμβάνει υπόψιν τις πολυπλοκότητες των δεδομένων.
3. **Ελαχιστοποίηση Πρόσθετων Παραμέτρων:** Θα εισάγει ελάχιστο αριθμό επιπλέον παραμέτρων στο προ-εκπαιδευμένο μοντέλο, διασφαλίζοντας ότι το επιπλέον υπολογιστικό φορτίο είναι χαμηλό.
4. **Βελτίωση Απόδοσης:** Θα υποδείξει, ύστερα από σύγκριση με σχετικές εργασίες, ότι οδηγεί σε βελτιώσεις στην ακρίβεια των κατάντη εργασιών.

1.3 Δομή Εργασίας

Η δομή αυτής της εργασίας οργανώνεται ως εξής:

- **Κεφάλαιο 1:** Πραγματοποιείται αναφορά στο κίνητρο ανάπτυξης της μεθόδους αυτής καθώς και οι στόχοι της
- **Κεφάλαιο 2:** Παρουσιάζονται σημαντικοί ορισμοί και θεωρία σχετικά με την Μηχανική και Βαθιά Μάθηση
- **Κεφάλαιο 3:** Περιγράφονται σημαντικές έννοιες για την προτεινόμενη μέθοδο όπως η αυτο-επιβλεπόμενη μάθηση και η αρχιτεκτονικές των Transformers και Vision Transformers
- **Κεφάλαιο 4:** Παρουσιάζονται σχετικές εργασίες στην βιβλιογραφία και πλαίσια (frameworks) από τα οποία θα αναπτυχθεί η παρούσα εργασία
- **Κεφάλαιο 5:** Περιγράφεται η μέθοδος, τα τεχνικά χαρακτηριστικά των πειραμάτων που πραγματοποιήθηκαν καθώς και ο εκτενής σχολιασμός των αποτελεσμάτων
- **Κεφάλαιο 6:** Περιγράφονται τα συμπεράσματα και οι μελλοντικές κατευθύνσεις

ΚΕΦΑΛΑΙΟ 2

Θεωρητικό Υπόβαθρο

2.1 Βασικές Έννοιες

Στην ενότητα αυτή πραγματοποιείται περιγραφή βασικών εννοιών που θα βοηθήσουν στην πλήρη κατανόηση της εργασίας και των μεθόδων που αναλύονται.

2.1.1 Μηχανική Μάθηση (Machine Learning)

Η Μηχανική Μάθηση είναι κομμάτι της τεχνητής νοημοσύνης και κατ'επέκταση της επιστήμης των υπολογιστών, η οποία εστιάζει στο να χρησιμοποιεί δεδομένα και να αναπτύσσει αλγορίθμους ώστε να «μαθαίνει» χωρίς συγκεκριμένους κανόνες, με τρόπο που μιμείται τον ανθρώπινο εγκέφαλο. Μέσω της χρήσης πολλών μεθόδων, κυρίως στατιστικής, εκπαιδεύονται οι αλγόριθμοι ώστε να κάνουν ταξινομήσεις και προβλέψεις οι οποίες θα αποβούν αργότερα σε λήψη αποφάσεων σε εφαρμογές και προβλήματα.

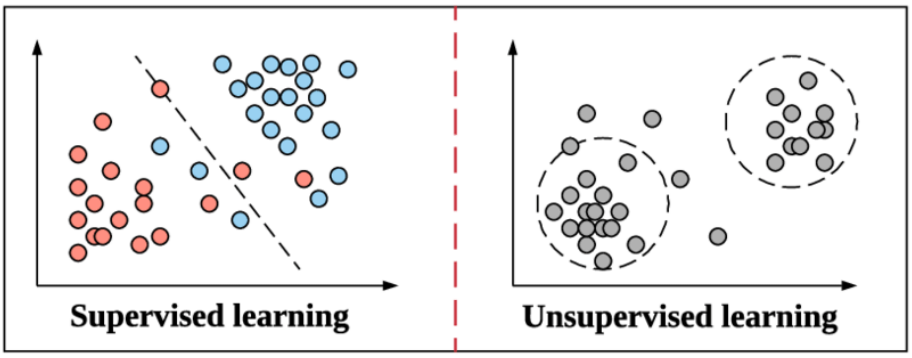
Η μηχανική μάθηση χωρίζεται σε τέσσερις κύριες κατηγορίες ανάλογα την φύση και τη δομή των δεδομένων εκπαίδευσης ή την επίβλεψη του μοντέλου:

- **Επιβλεπόμενη Μάθηση (supervised learning)**
- **Μη-Επιβλεπόμενη Μάθηση (unsupervised learning)**
- **Ημι-Επιβλεπόμενη Μάθηση (semi-supervised learning)**
- **Ενισχυμένη Μάθηση (reinforcement learning)**

Η **επιβλεπόμενη μάθηση** καθορίζεται από την χρήση επισημασμένων δεδομένων στα οποία θα εκπαιδευτεί το μοντέλο στοχεύοντας στην μάθηση κανόνων/συναρτήσεων

που συνδέουν τα αρχικά δεδομένα με τα επιθυμητά αποτελέσματα. Κάποιες βασικές εφαρμογές της επιβλεπόμενης μάθησης είναι η παλινδρόμηση, ταξινόμηση και η πρόβλεψη αποτελεσμάτων.

Στην **μη-επιβλεπόμενη μάθηση** ο υπολογιστής καλείται να αναλύσει και να κατηγοριοποιήσει μη-επισημασμένα δεδομένα που παίρνει ως είσοδο. Μια βασική εφαρμογή της μη-επιβλεπόμενης μάθησης είναι η μέθοδος ομαδοποίησης (clustering).



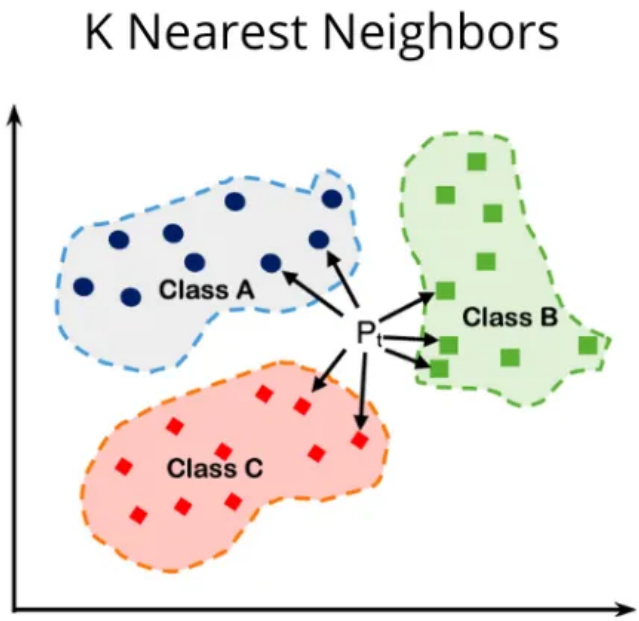
Σχήμα 2.1: Επιβλεπόμενη και Μη-Επιβλεπόμενη Μάθηση - *Πηγή*

Η **ημι-βλεπόμενη μάθηση** είναι ο συνδυασμός της επιβλεπόμενης με την μη-επιβλεπόμενη μάθηση. Στην ημι-επιβλεπόμενη μάθηση, δίνεται ως είσοδος ένα σετ δεδομένων με τα περισσότερα από αυτά να είναι μη-επισημασμένα και ένα μικρό ποσοστό επισημασμένα. Το μοντέλο γνωρίζει δηλαδή, ένα μικρό μέρος των επιθυμητών αποτελεσμάτων.

Στην **ενισχυμένη μάθηση** αναπτύσσεται ένα μοντέλο παρόμοιο με αυτό της επιβλεπόμενης μάθησης αλλά ο αλγόριθμος δεν εκπαιδεύεται σε επισημασμένα δεδομένα εισόδου και εξόδου. Αντιθέτως, αλληλεπιδρά σε ένα «εικονικό» περιβάλλον που έχει κατασκευαστεί το οποίο έχει συγκεκριμένους κανόνες και στόχους. Ο στόχος επιτυγχάνεται μέσω μιας διαδικασίας δοκιμών και λαθών. Η ενισχυμένη μάθηση, συχνότερα συναντάται στην πλοήγηση, στα παιχνίδια και στην ρομποτική.

Αξιολόγηση με -Εγγύτερους Γείτονες (k-NN) Η αξιολόγηση με k-εγγύτερους γείτονες (k-NN) είναι μια απλή αλλά αποτελεσματική μέθοδος για την αξιολόγηση της ποιότητας των χαρακτηριστικών που μαθαίνονται από ένα προ-εκπαιδευμένο μοντέλο. Σε αυτή τη μέθοδο, η ταξινόμηση ενός δείγματος καθορίζεται από την πλειοψηφία της κλάσης μεταξύ των k πλησιέστερων γειτόνων του στον χώρο των χαρακτηριστικών. Ακολουθεί τα εξής απλά βήματα:

1. **Προ-εκπαίδευση:** Εκπαίδευση μοντέλου με αυτό-επιβλεπόμενο τρόπο
2. **Εξαγωγή Χαρακτηριστικών:** Χρήση του προ-εκπαιδευμένου μοντέλου για την εξαγωγή χαρακτηριστικών από τα δεδομένα
3. **Ταξινόμηση με k-NN:** Για κάθε δείγμα δοκιμής, εντοπίζονται οι k πλησιέστεροι γείτονες από το σύνολο εκπαίδευσης και αποδίδεται η πλειοψηφούσα κλάση μεταξύ αυτών των γειτόνων



Σχήμα 2.2: Μέθοδος k-NN - *Πηγή*

Πλεονεκτήματα και Μειονεκτήματα Μεθόδου k-NN

Είναι μια μη παραμετρική μέθοδος και απλή στην υλοποίηση της και δεν απαιτεί πρόσθετη εκπαίδευση, καθιστώντας την υπολογιστικά αποδοτική. Ωστόσο, η απόδοση της εξαρτάται σημαντικά από την επιλογή του k και την απόσταση μέτρησης. Μπορεί επίσης να είναι αργή και απαιτητική σε μνήμη για μεγάλα σύνολα δεδομένων.

2.1.2 Βαθιά Μάθηση (Deep Learning)

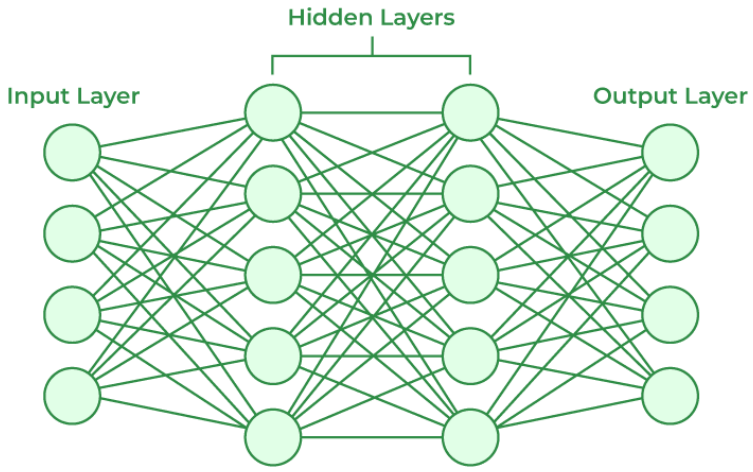
Η Βαθιά Μάθηση αποτελεί υποκατηγορία της Μηχανικής Μάθησης και τα κύρια χαρακτηριστικά της εντάσσονται στην ανάπτυξη πολυπλοκότερων και «βαθύτερων» μοντέλων με πολλά στρώματα με σκοπό την εξαγωγή χαρακτηριστικών υψηλότερου επιπέδου. Η τεχνική της βαθιάς μάθησης επινοήθηκε έχοντας πρότυπο την επεξεργασία

των οπτικών πληροφοριών που πραγματοποιεί ο ανθρώπινος εγκέφαλος μέσω των νευρώνων.

2.1.3 Νευρωνικά Δίκτυα (Neural Networks)

Τα νευρωνικά δίκτυα αποτελούν θεμελιώδη στοιχείο της τεχνητής νοημοσύνης και είναι βασικό χαρακτηριστικό της βαθιάς μάθησης. Έχουν εμπνευστεί από τον ανθρώπινο εγκέφαλο καθώς αποτελούνται από διασυνδεδεμένους κόμβους ή νευρώνες που οργανώνονται σε στρώματα (layers). Η αρχιτεκτονική τους διαμορφώνεται πρώτα από το στρώμα εισόδου στο οποίο δίνονται τα αρχικά δεδομένα, ύστερα από ένα ή περισσότερα ενδιάμεσα (κρυμμένα) στρώματα τα οποία επεξεργάζονται και μεταμορφώνουν τα δεδομένα μέσω σταθμισμένων διασυνδέσεων και τέλος, ένα στρώμα εξόδου που παράγει τα αποτελέσματα/προβλέψεις. Στη συνέχεια, θα αναλυθούν οι παρακάτω υποενότητες που αφορούν τα στρώματα ενός νευρωνικού δικτύου.

- **Συνελικτικά Στρώματα (convolutional layers)**
- **Επίπεδα Συγκέντρωσης (pooling layers)**
- **Πλήρη Συνδεδεμένα Στρώματα (fully connected layers)**

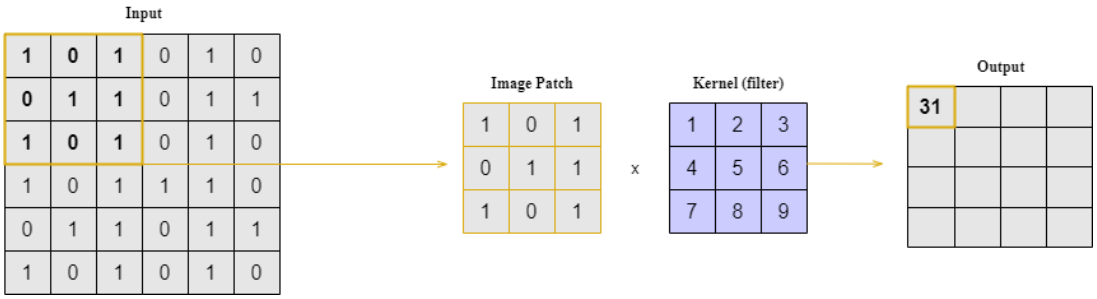


Σχήμα 2.3: Αρχιτεκτονική Νευρωνικών Δικτύων - *Πηγή*

Συνελικτικά Στρώματα (Convolutional Layers)

Τα συνελικτικά στρώματα έχουν σχεδιαστεί για να επεξεργάζονται δεδομένα σε μορφή πλέγματος, όπως οι εικόνες. Τα στρώματα αυτά, αποτελούνται από ένα σύνολο

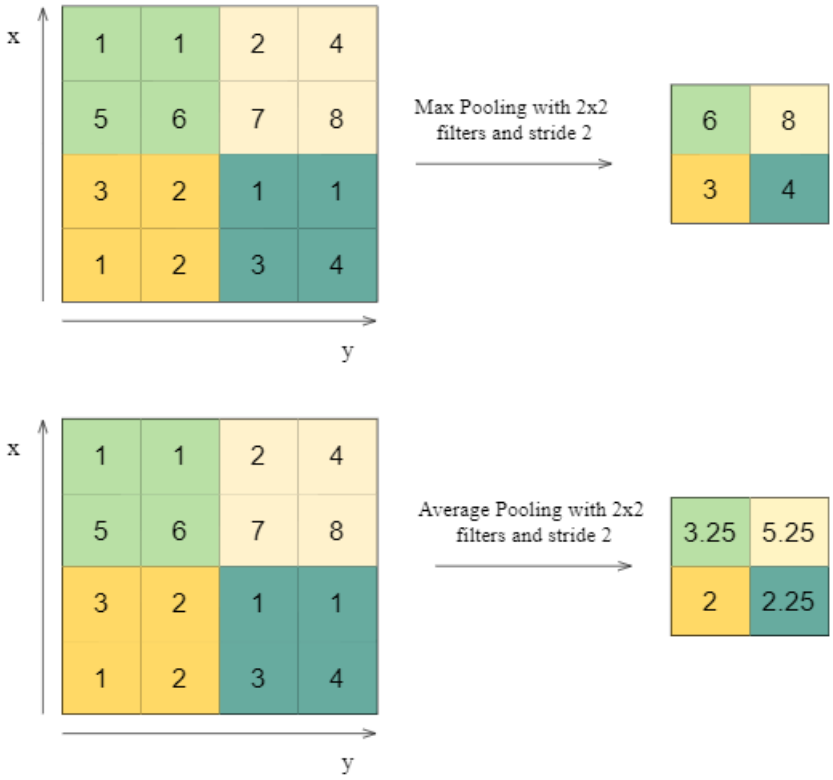
φίλτρων με δυνατότητα εκμάθησης, τα οποία εφαρμόζονται στα δεδομένα εισόδου και μεταξύ αυτών πραγματοποιείται μια πράξη, γνωστή και ως συνέλιξη, με σκοπό την εξαγωγή χαρακτηριστικών μοτίβων της εικόνας (feature maps). Συγκεκριμένα, κατά τη διαδικασία εκπαίδευσης του δικτύου, τα φίλτρα εξελίσσονται για να ανιχνεύουν χαρακτηριστικά όπως άκρες, υφές και πολύπλοκα μοτίβα. Κάθε φίλτρο αποτελεί ένα διδιάστατο πίνακα μεγέθους 3×3 ή 5×5, στην πλειοψηφία, ο οποίος εφαρμόζεται στην εικόνα και υπολογίζεται το εσωτερικό γινόμενο μεταξύ των εικονοστοιχείων και των βαρών που απαρτίζουν το φίλτρο. Πραγματοποιείται αρκετές φορές αυτή η διαδικασία, με τη μετάθεση του φίλτρου με καθορισμένο βήμα, ώστε να καλυφθούν όλα τα εικονοστοιχεία της εικόνας **(σχήμα 2.4)**. Το κοινό σχήμα βάρους εντός των συνελικτικών στρωμάτων, καθιστά την τεχνική αυτή υπολογιστικά αποδοτική τόσο σε χρόνο αλλά και απόδοση στην ταξινόμηση εικόνων, ανίχνευση αντικειμένων και τμηματοποίηση της εικόνας.



Σχήμα 2.4: Πράξη Συνέλιξης

Στρώματα Συγκέντρωσης (Pooling Layers)

Τα στρώματα συγκέντρωσης αποτελούν πολύ σημαντικά στοιχεία στα νευρωνικά δίκτυα. Ο κύριος ρόλος τους είναι να μειώσουν την διαστατικότητα των εισόδων, δηλαδή των χαρακτηριστικών που παράγονται από τα συνελικτικά στρώματα. Ένα στρώμα συγκέντρωσης λειτουργεί παρόμοια με ένα φίλτρο το οποίο εφαρμόζεται στα χαρακτηριστικά μοτίβα (feature maps) που έχουν εξαχθεί από τα συνελικτικά στρώματα και μπορεί να επιλεχθεί, ανάλογα την εφαρμογή, να πραγματοποιηθεί είτε μέγιστη επιλογή (*max pooling*) είτε μέση επιλογή (*average pooling*) (2.5). Όλη αυτή η διαδικασία βοηθά στην μείωση του υπολογιστικού φόρτου και του αριθμού των παραμέτρων που απαιτούνται και έτσι οι αναπαραστάσεις γίνονται μικρότερες και περισσότερο διαχειρίσιμες.



Σχήμα 2.5: Μέθοδοι Σγκέντρωσης (pooling methods)

Πλήρη Συνδεδεμένα Στρώματα (Fully Connected Layers)

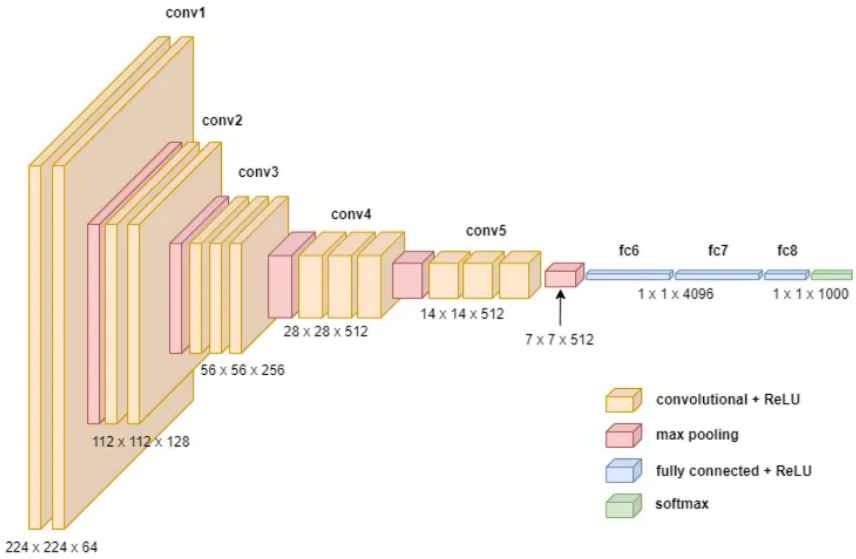
Τα πλήρη συνδεδεμένα στρώματα ενός νευρωνικού δικτύου εμπρόσθιας τροφοδότησης (feedforward network) ή ενός MLP (multilayer perceptron) δικτύου, χρησιμοποιούνται συνήθως στο τέλος του δικτύου, αφού έχουν προηγηθεί τα συνελικτικά στρώματα και η συγκέντρωση, με σκοπό την μετατροπή των χαρακτηριστικών μοτίβων (feature maps) σε κατανομή πιθανοτήτων κλάσεων. Κάθε νευρώνας ή κόμβος συνδέεται με κάθε νευρώνα από το προηγούμενο στρώμα και κάθε σύνδεση συσχετίζεται με μια παράμετρο βάρους την οποία μαθαίνει το δίκτυο κατά την διαδικασία της εκπαίδευσης μέσω συγκεκριμένων τεχνικών. Μαθηματικά, η έξοδος y ενός πλήρως συνδεδεμένου στρώματος μπορεί να υπολογιστεί ως εξής:

$$y = f(Wx + b) \tag{2.1}$$

όπου:

- W είναι ο πίνακας βαρών που συνδέει τους νευρώνες του προηγούμενου στρώματος με το πλήρως συνδεδεμένο στρώμα
- x είναι το διάνυσμα εισόδου από το προηγούμενο στρώμα

- b είναι το διάνυσμα μετατόπισης (bias)
- f είναι η μη γραμμική συνάρτηση ενεργοποίησης (activation function)



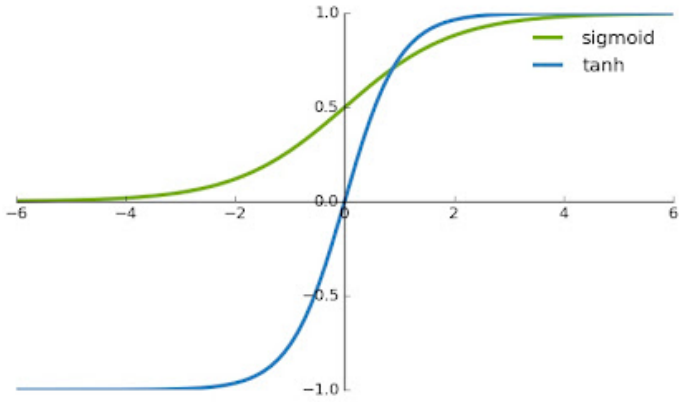
Σχήμα 2.6: Αρχιτεκτονική του VGG-Net (συνελικτικό δίκτυο) - *Πηγή*

2.1.4 Συναρτήσεις Ενεργοποίησης (Activation Functions)

Οι συναρτήσεις ενεργοποίησης αποτελούν σημαντική συνιστώσα για τα νευρωνικά δίκτυα καθώς εισάγουν μη-γραμμικότητες στο δίκτυο που του επιτρέπουν την εκμάθηση πολύπλοκων μοτίβων στα δεδομένα. Πιο συγκεκριμένα, αποφασίζουν εάν ένας νευρώνας πρέπει να ενεργοποιηθεί ή όχι, υπολογίζοντας το σταθμισμένο άθροισμα και προσθέτοντας μια εκπαιδεύσιμη παράμετρο (bias). Παρακάτω, παρουσιάζονται μερικές από τις πιο κοινές συναρτήσεις ενεργοποίησης:

- **Sigmoid Function:** Μια σιγμοειδής συνάρτηση εξάγει αποτελέσματα από 0 έως 1 ακολουθώντας την συνάρτηση $f(x) = \frac{1}{1+e^{-z}}$ και είναι χρήσιμη σε εφαρμογές πρόβλεψης πιθανότητας ως αποτέλεσμα και δυαδικής ταξινόμησης. Είναι μια διαφοροποίησιμη συνάρτηση στην οποία μπορεί να βρεθεί η κλίση της καμπύλης σε οποιαδήποτε δύο σημεία της. Ένα αρνητικό της συνάρτησης αυτής είναι ότι για ακραίες εισόδους, υποφέρει από το γνωστό πρόβλημα εξαφάνισης κλίσης (vanishing gradient) όπου οι κλίσεις γίνονται πολύ μικρές, οδηγώντας σε αργή εκμάθηση κατά την εκπαίδευση του δικτύου.

- **Tanh Function (Hyperbolic Tangent Function):** Η υπερβολική συνάρτηση εφαπτομένης $f(x) = \frac{2}{1+e^{-2x}} - 1$ είναι παρόμοια με την σιγμοειδή συνάρτηση όμως έχει το πλεονέκτημα ότι το εύρος τιμών που λαμβάνει είναι από -1 έως 1, οι αρνητικές είσοδοι θα αποδίδονται στις αρνητικές τιμές εξόδου και οι θετικές είσοδοι στις θετικές τιμές εξόδου. Είναι επίσης μια διαφοροποιήσιμη μονότονη συνάρτηση και χρησιμοποιείται σε εφαρμογές δυαδικής ταξινόμησης.

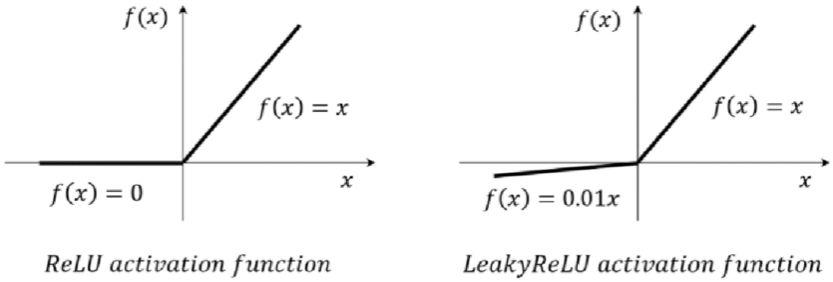


Σχήμα 2.7: (a) Σιγμοειδής Συνάρτηση, (b) Υπερβολική Συνάρτηση Εφαπτομένης - *Πηγή*

- **ReLU (Rectified Linear Unit):** Η συνάρτηση αυτή επιστρέφει την τιμή 0 αν η είσοδος της είναι αρνητική, ενώ για οποιαδήποτε θετική τιμή x επιστρέφει την ίδια τιμή x, ακολουθώντας την εξίσωση $f(x) = max(0, x)$. Το εύρος των τιμών της είναι $[0, +\infty]$. Η συνάρτηση ReLU προτιμάται συνήθως στα περισσότερα νευρωνικά δίκτυα λόγω της απλότητας καθώς και της αποτελεσματικότητας που προσδίδει στον μετριασμό του προβλήματος της εξαφάνισης της κλίσης (vanishing gradient). Ο βασικός περιορισμός της είναι πως για κάθε αρνητική τιμή εισόδου η κλίση μηδενίζεται, γεγονός το οποίο αποτρέπει τα βάρη να ανανεωθούν και να προσαρμοστούν κατάλληλα, οπότε και οι αντίστοιχοι νευρώνες αδρανοποιούνται πλήρως.

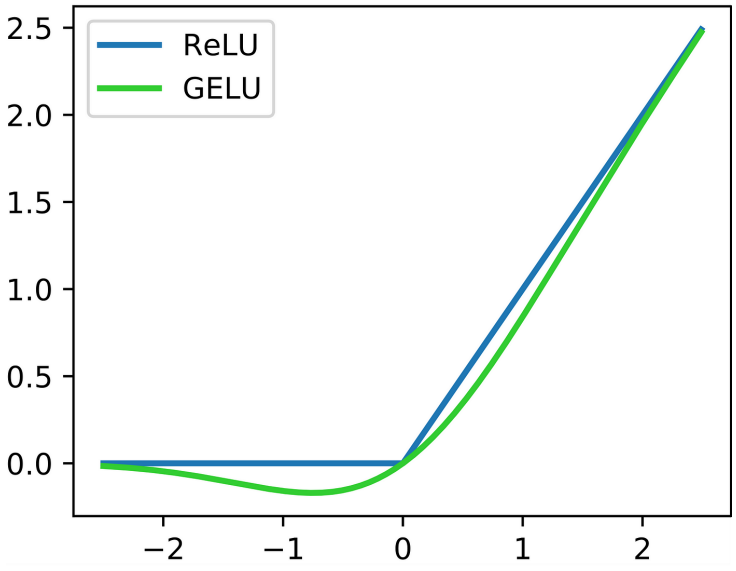
- **Leaky ReLU:** Αυτή η συνάρτηση είναι μια παραλλαγή της συνάρτησης ReLU, με την διαφορά ότι διορθώνει ως έναν βαθμό το πρόβλημα των αδρανών νευρώνων, εφαρμόζοντας μια μικρή σταθερή θετική κλίση. Ακολουθεί την εξίσωση $f(x) = max(ax, x)$ όπου συνήθως $a = 0.01$. Το εύρος των τιμών της είναι $[-\infty, +\infty]$ και έτσι κάθε αρνητική είσοδος δεν οδηγεί την κλίση σε μηδενισμό, οπότε οι αντίστοιχοι νευρώνες δεν αδρανοποιούνται.

- **GeLU (Gaussian Error Linear Unit):** Η συνάρτηση GeLU [2] αποτελεί μια πρόσφατη προσθήκη στην οικογένεια των συναρτήσεων ενεργοποίησης και έχει α-



Σχήμα 2.8: Συνάρτηση ενεργοποίησης ReLU (αριστερά), συνάρτηση ενεργοποίησης Leaky ReLU (δεξιά) - *Πηγή*

ποδειχθεί ότι βελτιώνει την απόδοση σε εφαρμογές όρασης υπολογιστών, επεξεργασία φυσικής γλώσσας και αναγνώριση φωνής. Χαρακτηρίζεται ως μια εξομάλυνση της συνάρτησης ReLU καθώς επιτρέπει μικρές αρνητικές τιμές όταν η είσοδος είναι μικρότερη του μηδενός. Η GeLU ακολουθεί την εξίσωση $f(x) = 0.5x\left(1 + \tanh\left(\sqrt{\frac{2}{\pi}}(x + 0.044715x^3)\right)\right)$

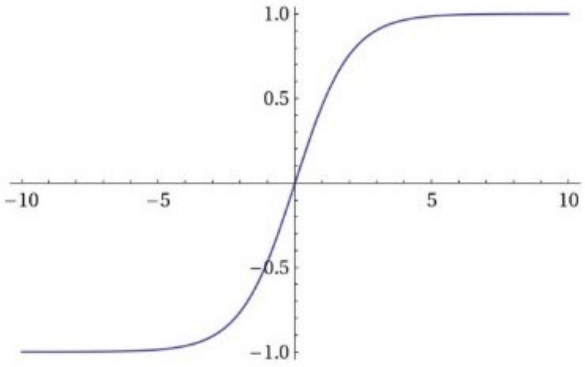


Σχήμα 2.9: (a) Συνάρτηση GeLU, (b) Συνάρτηση ReLU - *Πηγή*

- **Softmax:** Η συνάρτηση Softmax χρησιμοποιείται συνήθως σε προβλήματα ταξινόμησης πολλαπλών κατηγοριών. Μετατρέπει τις τελικές αριθμητικές εξόδους του μοντέλου σε μια κατανομή πιθανοτήτων $\sigma \in [0, 1]$ για κάθε πιθανή κατηγορία του. Η συνάρτηση ορίζεται από την εξίσωση:

$$\sigma(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{N} e^{z_j}} \tag{2.2}$$

όπου z_i το διάνυσμα εισόδου, z_j το διάνυσμα εξόδου και N το πλήθος των πιθανών κατηγοριών.



Σχήμα 2.10: Συνάρτηση Softmax - *Πηγή*

2.1.5 Συναρτήσεις Απώλειας (Loss Functions)

Οι συναρτήσεις απώλειας στα νευρωνικά δίκτυα μαζί με τους αλγόριθμους βελτιστοποίησης, που θα παρουσιαστούν εκτενώς στο επόμενο υποκεφάλαιο, αποτελούν σημαντικά στοιχεία για την προσαρμογή του μοντέλου στα δεδομένα εκπαίδευσης. Η συνάρτηση απώλειας συγκρίνει την πραγματική τιμή με την προβλεπόμενη τιμή του μοντέλου και στόχος είναι, κατά τη διάρκεια της εκπαίδευσης, να ελαχιστοποιείται. Επιπλέον, καθοδηγούν το μοντέλο στην βελτίωση του κατευθύνοντας τον αλγόριθμο να προσαρμόζει τις παραμέτρους (βάρη) επαναληπτικά, για να μειωθεί η απώλεια και να βελτιωθούν οι προβλέψεις. Μια από τις πιο γνωστές και συχνά χρησιμοποιούμενες συναρτήσεις απώλειας είναι η **διασταυρούμενη εντροπία** (cross entropy) ή αλλιώς λογαριθμική απώλεια (logarithmin loss) και χρησιμοποιείται σε εφαρμογές ταξινόμησης πολλών κλάσεων.

$$L(y, p) = -\sum_{i=1}^{N} y_i \cdot \log(p_i) \tag{2.3}$$

- N είναι ο αριθμός των κλάσεων
- y_i είναι ένας δυαδικός δείκτης (0 ή 1) εάν η κλάση i είναι η σωστή ταξινόμηση για ένα δείγμα
- p_i είναι η προβλεπόμενη πιθανότητα ότι το δείγμα ανήκει στην κλάση i.

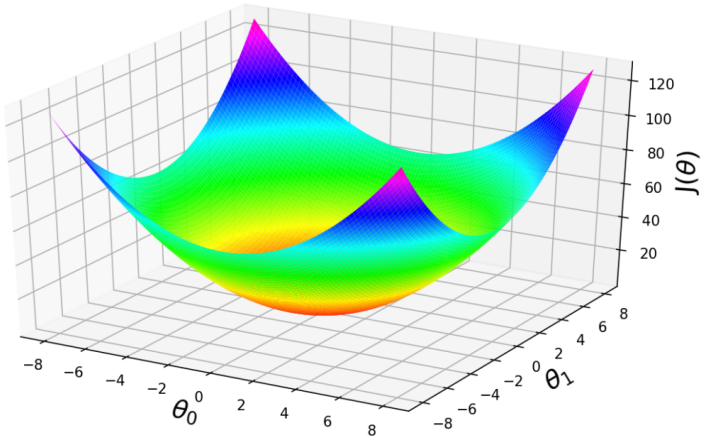
2.1.6 Αλγόριθμοι Βελτιστοποίησης (Optimization Algorithms)

Οι αλγόριθμοι βελτιστοποίησης χρησιμοποιούνται για να προσαρμόζουν τα χαρακτηριστικά του δικτύου όπως τα βάρη και τον ρυθμό εκμάθησης κατά την εκπαίδευση, προκειμένου να ελαχιστοποιηθούν οι απώλειες. Υπάρχουν πολλοί αλγόριθμοι βελτιστοποίησης, με διαφορετικά πλεονεκτήματα και αδυναμίες ο κάθε ένας, και η επιλογή τους πραγματοποιείται με βάση τα προβλήματα και τις αρχιτεκτονικές που συναντώνται. Μερικοί από τους πιο συχνά επιλεγόμενους αλγορίθμους είναι οι εξής:

- **Κάθοδος Κλίσης:** Είναι ένας αλγόριθμος βελτιστοποίησης πρώτης τάξης και εξαρτάται από την παράγωγο πρώτης τάξης της συνάρτησης απώλειας. Υπολογίζει με ποιον τρόπο πρέπει να προσαρμοστούν τα βάρη ώστε η συνάρτηση να φτάσει στο ελάχιστο. Χαρακτηρίζεται για την απλότητα υλοποίησης του, αλλά συγχρόνως εμφανίζει και ορισμένα αρνητικά στοιχεία, όπως το γεγονός ότι, για ένα πολύ μεγάλο σύνολο δεδομένων, θα χρειαστεί πάρα πολύ καιρό μέχρι να συγκλίνει στο ελάχιστο και απαιτεί επίσης μεγάλη μνήμη για τον υπολογισμό του.

$$\theta = \theta - \alpha \cdot \nabla J(\theta) \tag{2.4}$$

  - θ αντιπροσωπεύει τον αριθμό των παραμέτρων
  - α είναι ο ρυθμός εκμάθησης
  - $J(\theta)$ η συνάρτηση απώλειας
  - $\nabla J(\theta)$ υποδηλώνει την κλίση της συνάρτησης κόστους J ως προς τις παραμέτρους του θ

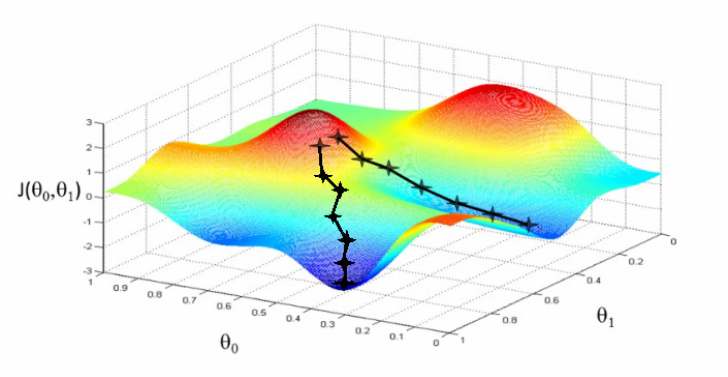


Σχήμα 2.11: Gradient Descent - *Πηγή*

- **Στοχαστική Κάθοδος Κλίσης:** Η στοχαστική κάθοδος κλίσης (Stochastic Gradient Descent-SGD) είναι ένας αλγόριθμος βελτιστοποίησης και αφορά μια παραλλαγή της καθόδου κλίσης με την διαφορά ότι ενημερώνει τις παραμέτρους του μοντέλου πιο συχνά. Έτσι, εάν το σύνολο δεδομένων περιέχει 100 γραμμές, ο αλγόριθμος αυτός θα ενημερώσει τις παραμέτρους 100 φορές σε ένα κύκλο των δεδομένων και όχι μία φορά όπως στην κάθοδο κλίσης. Τα πλεονεκτήματα του είναι πως συγκλίνει γρηγορότερα και απαιτεί λιγότερη μνήμη κατά τον υπολογισμό. Στα αρνητικά καταλογίζονται η υψηλή διακύμανση των παραμέτρων του μοντέλου και η αργή μείωση του ρυθμού εκμάθησης ώστε να φτάσει σε ίδια σύγκλιση με τον κάθοδο κλίσης.

$$\theta_j = \theta_j - \alpha \cdot \nabla J(\theta; x^{(i)}, y^{(i)}) \tag{2.5}$$

  - θ αντιπροσωπεύει τον αριθμό των παραμέτρων
  - α είναι ο ρυθμός εκμάθησης
  - $J(\theta)$ η συνάρτηση απώλειας
  - $\nabla J(\theta; x^{(i)}, y^{(i)})$ υποδηλώνει την κλίση της συνάρτησης απώλειας J ως προς τις παραμέτρους του θ που υπολογίζονται για ένα μόνο σημείο δεδομένων στην συγκεκριμένη μικρή παρτίδα (mini-batch)



Σχήμα 2.12: Στοχαστική Κάθοδος Κλίσης - *Πηγή*

- **Προσαρμοστική Εκτίμηση Ροπής:** Η προσαρμοστική εκτίμηση ροπής (Adaptive Moment Estimation-ADAM) [4] λειτουργεί με ορμές (momentums) πρώτης και δεύτερης τάξης με σκοπό να προσαρμόσει τον ρυθμό εκμάθησης για κάθε παράμετρο. Το σκεπτικό πίσω από αυτή τη προσέγγιση είναι η επίτευξη πιο σταθερής και αποτελεσματικής βελτιστοποίησης σε σύγκριση με μεθόδους που χρησιμοποιούν σταθερό ρυθμό εκμάθησης. Διατηρώντας εκθετικά φθίνοντες μέσους

όρους των περασμένων κλίσεων (πρώτης τάξης) και περασμένων τετραγωνικών κλίσεων (δεύτερης τάξης), ο αλγόριθμος αυτός μπορεί να κλιμακώσει προσαρμοστικά τους ρυθμούς εκμάθησης για διαφορετικές παραμέτρους με βάση την ιστορική τους συμπεριφορά κατά την βελτιστοποίηση. Αυτό βοηθά καθοριστικά στον χειρισμό αραιών κλίσεων, μην σταθερών στόχων και θορυβωδών κλίσεων. Αποτελεί μια πολύ γρήγορη μέθοδο με γρήγορο ρυθμό σύγκλισης όμως είναι υπολογιστικά δαπανηρή.

$$\hat{m} = \frac{m_t}{1 - \beta_1^t} \tag{2.6}$$

$$\hat{v} = \frac{u_t}{1 - \beta_2^t} \tag{2.7}$$

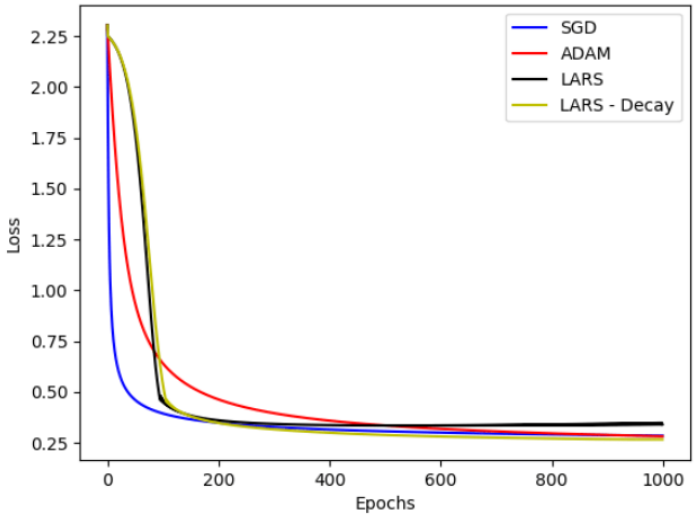
  - m_t είναι η αρχική ορμή
  - u_t είναι η εκτίμηση διακύμανσης
  - β_1, β_2 είναι υπερπαράμετροι
  - ο εκθέτης ^{t} αντιπροσωπεύει το τρέχον χρονικό βήμα

- **Προσαρμοστική Κλιμάκωση Ρυθμού ανά Στρώμα:** Η προσαρμοστική κλιμάκωση ρυθμού ανα στρώμα (Layer-wise Adaptive Rate Scaling-LARS)[6], αποτελεί έναν αλγόριθμο βελτιστοποίησης και είναι μια παραλλαγή της στοχαστικής καθόδου κλίσης, ειδικά σχεδιασμένος για να βελτιώνει την απόδοση της εκπαίδευσης σε μεγάλες παρτίδες δεδομένων. Αντιμετωπίζει σε σημαντικό βαθμό ορισμένα από τα βασικά προβλήματα που υπάρχουν σε τέτοιες περιπτώσεις όπως η αργή σύγκλιση και η υποβέλτιστη απόδοση. Παρακάτω, παρουσιάζονται ορισμένα από τα βασικά του χαρακτηριστικά:

  Κλιμάκωση Ρυθμού Εκμάθησης ανά Στρώμα: Ο LARS κλιμακώνει τον ρυθμό εκμάθησης για κάθε Στρώμα (layer) ξεχωριστά βασιζόμενος στην νόρμα των βαρών και την νόρμα των βαθμίδων του επιπέδου. Αυτό βοηθά στην διατήρηση μιας σταθερής διαδικασίας εκπαίδευσης, ειδικά όταν χρησιμοποιούνται μεγάλες παρτίδες.

$$\eta_l = \eta \times \frac{\|W_l\|}{\|G_l\|} \tag{2.8}$$

  - η είναι ο γενικός ρυθμός εκμάθησης
  - W_l είναι το διάνυσμα βαρών του επιπέδου l
  - G_l είναι το διάνυσμα βαθμίδων του επιπέδου l



Σχήμα 2.13: Σύγκριση Αλγορίθμων Βελτιστοποίησης SGD-ADAM-LARS - *Πηγή*

Στο σχήμα 2.13 παρουσιάζονται οι διαφορές μεταξύ των αλγορίθμων βελτιστοποίησης SGD-ADAM-LARS. Μακροπρόθεσμα, όλες οι καμπύλες απωλειών των αλγορίθμων ισοπεδώνονται, υποδεικνύοντας ότι το μοντέλο συγκλίνει σε μια ελάχιστη τιμή απώλειας. Ο LARS φαίνεται να διατηρεί χαμηλότερη απώλεια σε σύγκριση με SGD και ADAM, υποδηλώνοντας καλύτερη απόδοση σε μεγάλες παρτίδες δεδομένων.

ΚΕΦΑΛΑΙΟ 3

Αυτο-επιβλεπόμενη μάθηση και Μετασχηματιστές

Στο κεφάλαιο αυτό καλύπτεται το θεωρητικό υπόβαθρο πίσω από τις αρχιτεκτονικές βαθιών νευρωνικών δικτύων που χρησιμοποιούνται στο πλαίσιο της εργασίας. Σε πρώτο βήμα, περιγράφεται εκτενώς η έννοια της αυτο-επιβλεπόμενης μάθησης, με την οποία έχουν αναπτυχθεί τα πλαίσια που θα βασιστεί η παρούσα εργασία. Σε δεύτερη φάση, παρουσιάζεται όλη η θεωρία σχετικά με την αρχιτεκτονική των μοντέλων Transformer και τον μηχανισμό αυτο-προσοχής [5]. Τα μοντέλα αυτά και ο μηχανισμός αυτο-προσοχής έχουν αναπτυχθεί κυρίως για εφαρμογές επεξεργασίας φυσικής γλώσσας (natural language processing). Ωστόσο, η παρούσα εργασία εστιάζει σε εφαρμογές εικόνων, και επομένως, στο τελευταίο μέρος του κεφαλαίου γίνεται αναφορά στην αρχιτεκτονική του Vision Transformer [12], ο οποίος έχει αναπτυχθεί ειδικά για την επεξεργασία εικόνων.

3.1 Αυτο-επιβλεπόμενη Μάθηση (Self-supervised Leanring)

Η αυτο-επιβλεπόμενη μάθηση είναι μια εξελισσόμενη τεχνική στον τομέα της Μηχανικής Μάθησης, η οποία λύνει τις προκλήσεις που θέτει η υπερβολική εξάρτηση από δεδομένα με επισημασμένες γνωστές ετικέτες. Παραδοσιακά, οι εφαρμογές επιβλεπόμενης μάθησης απαιτούν τέτοιου είδους δεδομένα καλής ποιότητας τα οποία μπορεί να είναι δαπανηρά και χρονοβόρα για να αποκτηθούν σε μεγάλη κλίμακα. Σκοπός λοιπόν, ήταν να αναπτυχθούν αυτο-επιβλεπόμενοι μηχανισμοί με μη επισημασμένα δεδομένα τα οποία να μπορούν να κλιμακώσουν την έρευνα και την ανάπτυξη γενικών συστημάτων τεχνητής νοημοσύνης με χαμηλό κόστος. Ο σκοπός επιτεύχθηκε με την αυτο-επιβλεπόμενη μάθηση καθώς αποτελεί μια διαδικασία όπου το μοντέλο εκπαιδεύεται για να μάθει ένα μέρος της εισόδου των δεδομένων από ένα άλλο μέρος

δεδομένων εισόδου.

Διαφορά μη-επιβλεπόμενης και αυτο-επιβλεπόμενης μάθησης

Η μάθηση χωρίς επίβλεψη και η αυτο-επιβλεπόμενη μάθηση μπορούν να θεωρηθούν συμπληρωματικές τεχνικές καθώς και οι δύο δεν χρειάζονται δεδομένα με επισημασμένες ετικέτες. Η πρώτη, εστιάζει περισσότερο στο μοντέλο και όχι στα δεδομένα ενώ η τεχνική αυτο-επιβλεπόμενης μάθησης λειτουργεί αντίστροφα. Η μάθηση χωρίς επίβλεψη αποφέρει καλά αποτελέσματα σε εφαρμογές ομαδοποίησης (clustering) και μείωσης διαστάσεων, ενώ η αυτο-επιβλεπόμενη είναι μια προσχεδιασμένη μέθοδος για εφαρμογές παλινδρόμησης και ταξινόμησης.

Γιατί αυτό-επιβλεπόμενη μάθηση;

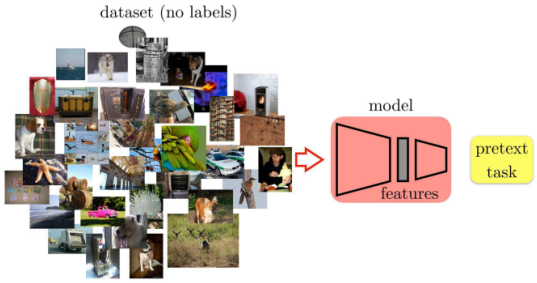
- **Τα δεδομένα με επισημασμένες ετικέτες κοστίζουν**: Η απόκτηση δεδομένων με ετικέτα μπορεί να είναι πολύ δαπανηρή, ειδικά όταν χρειάζονται μεγάλες ποσότητες υψηλής ποιότητας. Η αυτο-επιβλεπόμενη μάθηση επιτρέπει στα μοντέλα να εκπαιδεύονται σε τεράστιες ποσότητες δεδομένων χωρίς ετικέτα, μειώνοντας κατά κόρον την εξάρτηση από δαπανηρά σύνολα.
- **Αυτοματοποίηση**: Η προετοιμασία των δεδομένων για την επιβλεπόμενη μάθηση περιλαμβάνει πολλαπλά βήματα όπως καθαρισμό, φιλτράρισμα, σχολιασμό και μετασχηματισμό. Τα βήματα αυτά απαιτούν ανθρώπινη παρέμβαση και τεχνογνωσία, παρατείνοντας τον χρόνο που απαιτείται για την προετοιμασία τους. Η αυτο-επιβλεπόμενη μέθοδος απλοποιεί την παραπάνω διαδικασία αξιοποιώντας την εγγενή δομή των μη-επισημασμένων δεδομένων, μειώνοντας έτσι την ανάγκη για εκτενή χειροκίνητη προεπεξεργασία δεδομένων.
- **Ένα βήμα πιο κοντά στην ανθρώπινη αντίληψη:** Μιμείται πτυχές της ανθρώπινης γνώσης επιτρέποντας στα μοντέλα να μαθαίνουν τα δεδομένα χωρίς ρητή επίβλεψη. Επιπρόσθετα, μαθαίνουν να εξάγουν αναπαραστάσεις από τα ακατέργαστα δεδομένα, παρόμοια με το πως οι άνθρωποι αντιλαμβάνονται και εξάγουν πληροφορίες από το περιβάλλον τους.

Για την εκπαίδευση ενός αυτο-επιβλεπόμενου μοντέλου ακολουθούνται συνήθως δύο στάδια. Πρώτο στάδιο αποτελεί η προσχεδιασμένη εργασία ή αλλιώς **προεκπαίδευση**. Σκοπός αυτής της εργασίας είναι να καθοδηγήσει το μοντέλο για να μάθει ενδιάμεσες

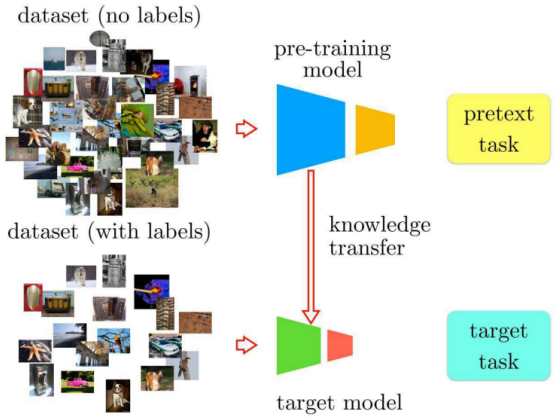
αναπαραστάσεις δεδομένων. Αυτό το στάδιο καθίστανται πολύ χρήσιμο για την κατανόηση της δομικής σημασίας των δεδομένων που θα επωφελήσει το επόμενο στάδιο που αφορά τις κατάντη εργασίες (**downstream tasks**). Μια κατάντη εργασία είναι η διαδικασία μεταφοράς γνώσης από το πρώτο στάδιο σε μια συγκεκριμένη εργασία και τροφοδοτείται με μικρότερη ποσότητα επισημασμένων δεδομένων. Η εργασία αυτή μπορεί να αφορά ταξινόμηση αντικειμένων, αναγνώριση αντικειμένων καθώς και άλλες εφαρμογές και τελικώς επανεκπαιδεύεται από την προσχεδιασμένη εργασία.

Η εξέλιξη των προσχεδιασμένων εργασιών ξεκίνησε από απλά προβλήματα όπως η περιστροφή εικόνων (rotations) [7] και παζλ (jigsaw) [3] , όπου το μοντέλο έπρεπε να μάθει να αναγνωρίζει τη σωστή γωνία περιστροφής μιας εικόνας ή να συναρμολογεί κομμάτια ενός παζλ από εικόνες. Αυτές οι μέθοδοι προσέφεραν ένα αρχικό βήμα στην κατανόηση των βασικών δομών των δεδομένων. Στη συνέχεια, οι τεχνικές αυτές εξελίχθηκαν σε πιο προηγμένες μεθόδους όπως η αντιθετική μάθηση (contrastive learning) [10], όπου το μοντέλο μαθαίνει να διακρίνει μεταξύ παρόμοιων και μη παρόμοιων παραδειγμάτων. Αυτή η προσέγγιση οδήγησε σε καλύτερες αναπαραστάσεις δεδομένων και ήταν ένας κρίσιμος παράγοντας στην ανάπτυξη σύγχρονων μεθόδων αυτο-επιβλεπόμενης μάθησης.

Η έρευνα σε μεθόδους αυτο-επιβλεπόμενης μάθησης οδήγησε στη μέθοδο DINO (self-distillation with no labels), η οποία βασίζεται σε ένα δίκτυο δασκάλου-μαθητή (teacher-student network). Σε αυτό το δίκτυο, εκπαιδεύεται πρώτα ένα μοντέλου δασκάλου και στη συνέχεια χρησιμοποείται για να καθοδηγήσει την εκπαίδευση ενός δεύτερου μοντέλου (μαθητή) χωρίς τη χρήση επισημασμένων δεδομένων. Η μέθοδος iBoT [18] αποτελεί μια περαιτέρω εξέλιξη αυτής της προσέγγισης, ενσωματώνοντας μια επιπλέον συνάρτηση απώλειας η οποία βοηθά στην βελτίωση των αναπαραστάσεων και ενισχύει την ακρίβεια του μοντέλου σε κατάντη εργασίες. Παρόμοια τεχνική ακολούθησε μεταγενέστερα η ενημερωμένη μορφή του μοντέλου DINO, η DINOv2 [19]. Αξίζει να σημειωθεί πως πολλαπλές έρευνες υλοποιήθηκαν και συνεχίζουν να υλοποιούνται στα πλαίσια της βελτίωσης των εφαρμογών της αυτο-επιβλεπόμενη μάθηση. Η μέθοδος AttMask [16] αποτελεί μια τέτοια προσέγγιση, η οποία εισάγει μια στρατηγική προσεγγισμένης μάσκας για την αυτο-επιβλεπόμενη μάθηση, βελτιώνοντας την αποτελεσματικότητα σε διάφορες κατάντη εργασίες. Η μέθοδος MAE [15] αποτελεί επίσης μια εφαρμογή αυτο-επιβλεπόμενης μάθησης, η οποία αποτελεί μια απο τις τρεις μεθόδους που θα επικεντρωθεί η παρούσα διπλωματική εργασία και θα αναλυθούν περαιτέρω στα προσεχή κεφάλαια.



Σχήμα 3.1: Προσχεδιασμένη Εργασία (Pretext Task) - *Πηγή:[8]*



Σχήμα 3.2: Προσχεδιασμένη εργασία (pretext task) και κατάντη εργασία (downstream task). Οι περισσότερες τρέχουσες προσεγγίσεις αυτο-επιβλεπόμενης μάθησης χρησιμοποιούν την ίδια αρχιτεκτονική τόσο στην προ-εκπαίδευση όσο και στην βελτιωμένη ρύθμιση (fine-tuning). Αναπτύσσεται μια μέθοδος μεταφοράς γνώσης για την αποσύνδεση αυτών των δύο αρχιτεκτονικών. Αυτό επιτρέπει να χρησιμοποιείται ένα βαθύτερο μοντέλο στην προ-εκπαίδευση - *Πηγή:[8]*

3.2 Μετασχηματιστές

Σε αυτό το σημείο, πραγματοποιείται ανάλυση της αρχιτεκτονικής των μετασχηματιστών και του μηχανισμού αυτο-προσοχής. Πρώτα, θα μελετηθεί εις βάθος ο μηχανισμός αυτο-προσοχής, μια σημαντική καινοτομία στη σύγχρονη μοντελοποίηση ακολουθιών, ο οποίος επιτρέπει στο μοντέλο να εστιάζει δυναμικά σε διαφορετικά μέρη της ακολουθίας εισόδου, καταγράφοντας πολύπλοκες σχέσεις και μακροπρόθεσμες εξαρτήσεις στα δεδομένα. Σε επόμενο βήμα, θα αναλυθεί η αρχιτεκτονική μετασχηματιστών, η οποία ενσωματώνει στρώματα αυτο-προσοχής και νευρωνικά δίκτυα εμπρόσθιας τροφοδότησης (feedforward neural networks), ώστε να παρέχει ένα αποτελεσματικό και ευέλικτο τρόπο κατανόησης και επεξεργασίας σειριακών δεδομένων.

3.2.1 Αυτο-Προσοχή (Self-Attention)

Ο μηχανισμός αυτο-προσοχής ενισχύει την αναπαράσταση των ακολουθιών εισόδου, επιτρέποντας σε κάθε στοιχείο να αλληλεπιδρά με όλα τα υπόλοιπα στοιχεία. Αυτή η διαδικασία, καθορίζει σε ποια μέρη των πληροφοριών εισόδου θα εστιάσει περισσότερο και σε ποια λιγότερο, με βάση τη συνάφεια του πλαισίου. Όλα τα παραπάνω επιτυγχάνονται ύστερα από μια σειρά βημάτων, μέσω γραμμικών προβολών, υπολογισμού βαθμολογίας προσοχής (attention scores) και κανονικοποίηση softmax. Η τελική έξοδος είναι ένα σταθμισμένο άθροισμα των τιμών για κάθε είσοδο, όπου τα βάρη έχουν καθοριστεί από τη βαθμολογία προσοχής. Ακολούθως, πραγματοποιείται εκτενέστερη ανάλυση των βημάτων του μηχανισμού αυτο-προσοχής τα οποία συνολικά συνθέτουν το κλιμακωτό βάρος προσοχής.

Κλιμακωτό βάρος προσοχής (Scaled Dot-Product Attention)

Αρχικά, όλη η διαδικασία ξεκινά με μια είσοδο ακολουθίας που αποτελείται από n στοιχεία, το κάθε ένα με d διαστάσεις, σχηματίζοντας έναν πίνακα $X \in \mathbb{R}^{n \times d}$. Στη συνέχεια, πραγματοποιείται αρχικοποίηση των πινάκων βαρών $W_Q \in \mathbb{R}^{d \times d_q}$, $W_K \in \mathbb{R}^{d \times d_k}$ και $W_V \in \mathbb{R}^{d \times d_v}$. Το αμέσως επόμενο βήμα περιλαμβάνει τον υπολογισμό τριών σετ διανυσμάτων που αντιπροσωπεύουν τα ερωτήματα **(queries)** Q, κλειδιά **(keys)** K και τιμές **(values)** V, πολλαπλασιάζοντας τις εισόδους (πίνακας X) με τον αντίστοιχο πίνακα βαρών.

$$Q = X \cdot W_Q \tag{3.1}$$

$$K = X \cdot W_K \tag{3.2}$$

$$V = X \cdot W_V \tag{3.3}$$

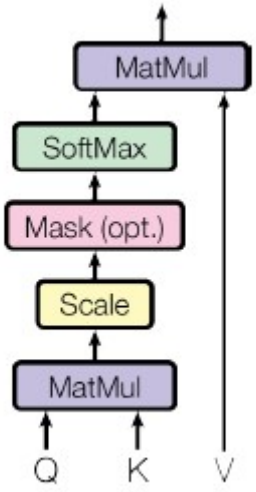
Όπου, $Q \in \mathbb{R}^{n \times d_q}$, $K \in \mathbb{R}^{n \times d_k}$, $V \in \mathbb{R}^{n \times d_v}$, και $d_q = d_k$. Αξίζει να σημειωθεί ότι η διάσταση d_v μπορεί να διαφέρει από τις άλλες δύο όμως πάντα αντιστοιχεί στη διάσταση εξόδου του μηχανισμού αυτο-προσοχής.

Στη συνέχεια, υπολογίζονται τα σκορ προσοχής με τον υπολογισμό του εσωτερικού γινομένου μεταξύ κάθε *μεμονωμένου* διανύσματος ερωτήματος Q και κάθε διανύσματος κλειδιού K. Με σκοπό να αποφευχθούν μεγάλα σκορ προσοχής, ύστερα από την εφαρμογή της Softmax, που οδηγούν σε μικρές τιμές, πραγματοποιείται κανονικοποίηση των σκορ διαιρώντας τα με την τετραγωνική ρίζα του d_k:

$$\frac{K \cdot Q^T}{\sqrt{d_k}} \tag{3.4}$$

Αμέσως μετά την κανονικοποίηση, εφαρμόζεται η συνάρτηση Softmax στα σκορ και έπειτα, τα αποτελέσματα αυτής της πράξης πολλαπλασιάζονται με τον πίνακα τιμών V ώστε να προκύψουν οι τιμές των βαρών προσοχής. Τέλος, οι τιμές των βαρών αθροίζονται στοιχείο προς στοιχείο (element-wise) με αποτέλεσμα την απόκτηση της τελικής εξόδου του μηχανισμού προσοχής. Αυτή η διαδικασία επαναλαμβάνεται για κάθε στοιχείο εισόδου. Παρακάτω, δίνεται η συνολική εξίσωση που χρησιμοποιεί ο μηχανισμός προσοχής.

$$Attention(Q, K, V) = softmax\left(\frac{K \cdot Q^T}{\sqrt{d_k}}\right) \cdot V \tag{3.5}$$



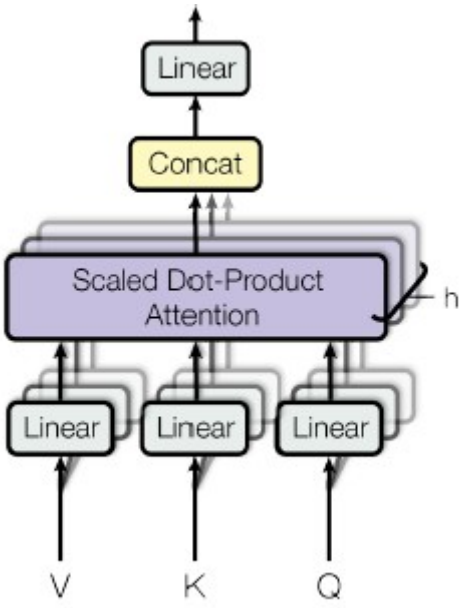
Σχήμα 3.3: Scaled Dot-Product Attention - *Πηγή:[5]*

Μηχανισμός Προσοχής Πολλαπλών Κεφαλιών

Κάθε μηχανισμός προσοχής υπολογίζει ένα σταθμισμένο μέσο όρο τιμών εισόδου σε ένα σταθμισμένο άθροισμα και το βάρος καθορίζεται από τις τιμές εισόδου. Η προσοχή πολλαπλών κεφαλιών μπορεί να θεωρηθεί ως η επέκταση αυτής της ιδέας, καθώς πολλαπλοί μηχανισμοί προσοχής λειτουργούν παράλληλα.

Πως λειτουργεί η προσοχή πολλαπλών κεφαλιών;

1. **Γραμμικές Παρεμβολές**: Κάθε *κεφάλι* έχει το δικό του σύνολο γραμμικών προβολών των ερωτημάτων Q, των κλειδιών K και των τιμών V. Αυτές οι προβολές μπορούν να θεωρηθούν ως διαφορετικοί υποχώροι στους οποίους υπολογίζεται η προσοχή.



Σχήμα 3.4: Multihead Attention - *Πηγή:[5]*

2. **Παράλληλα Κεφάλια Προσοχής**: Κάθε *κεφάλι* προσοχής λειτουργεί με τα ίδια δεδομένα εισόδου, αλλά χρησιμοποιεί διαφορετικές προβολές. Αυτό σημαίνει ότι κάθε ένα από αυτά μπορεί να εστιάσει σε διαφορετικά μέρη των δεδομένων, καταγράφοντας διαφορετικές σχέσεις.

3. **Τελικό Γραμμικό Στρώμα**: Οι έξοδοι των πολλαπλών κεφαλιών προσοχής συνενώνονται και στη συνέχεια προβάλλονται μέσω ενός τελικού γραμμικού στρώματος. Αυτός ο συνδυασμός επιτρέπει στο μοντέλο να ενσωματώσει τις πολλαπλές πληροφορίες που εξάγονται από αυτά.

Η μαθηματική διατύπωση της προσοχής πολλαπλών κεφαλιών δίνεται από την εξίσωση 3.6 όπου κάθε κεφάλι υπολογίζεται σύμφωνα με την εξίσωση 3.7.

$$MultiHead(Q, K, V) = concat(head_1, head_2, \ldots, head_n) \cdot W^O \tag{3.6}$$

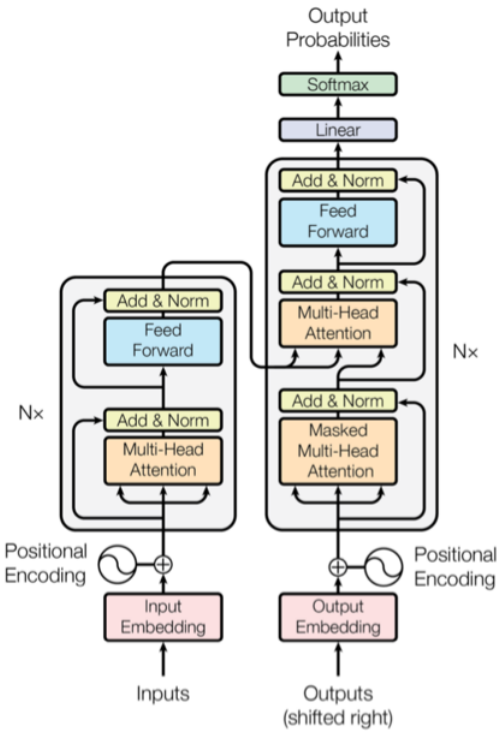
$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \tag{3.7}$$

3.2.2 Αρχιτεκτονική Μετασχηματιστών

Το μοντέλο μετασχηματιστή και ο μηχανισμός προσοχής που συναντάται στην αρχιτεκτονική του, εισήχθησαν το 2017 [5] ως μια νέα προσέγγιση σε εργασίες αλληλουχίας

σε ακολουθία. Αρχικά, προτάθηκε για αυτόματη μετάφραση κειμένου αλλά πλέον εφαρμόζεται σε μεγάλο εύρος εφαρμογών λόγω της ευελιξίας και της ισχυρής απόδοσης που το χαρακτηρίζει.

Η δομή της αρχιτεκτονικής αποτελείται από έναν κωδικοποιητή (encoder) και έναν αποκωδικοποιητή (decoder) και παρουσιάζεται στο σχήμα 3.5.



Σχήμα 3.5: Αρχιτεκτονική Μοντέλου Μετασχηματιστή - *Πηγή:[5]*

Κωδικοποιητής(Encoder)

Αποτελείται από N πανομοιότυπα στρώματα, το κάθε ένα με δύο κύρια υποστρώματα. Τα υποστρώματα περιέχουν **μηχανισμό πολλαπλών κεφαλιών** και ένα **πλήρως συνδεδεμένο δίκτυο εμπρόσθιας τροφοδοσίας** (1.2). Η προσοχή πολλαπλών κεφαλιών, όπως προαναφέρθηκε, επιτρέπει στον κωδικοποιητή να εστιάζει σε διαφορετικά μέρη της ακολουθίας εισόδου και να καταγράφει σημαντικές εξαρτήσεις μεταξύ αυτών ώστε τελικώς να έχει μια πλήρης κατανόηση της εισόδου. Το πλήρως συνδεδεμένο δίκτυο αποτελείται από δύο γραμμικά στρώματα και μια μη-γραμμική συνάρτηση ενεργοποίησης ανάμεσα τους. Για να βελτιωθεί η ροή των πληροφοριών και να διευκολυνθεί η εκπαίδευση του μοντέλου, κάθε υπόστρωμα περιλαμβάνει μια σχέση παράκαμψης (skip connection), γνωστή και ως υπολειμματική σύνδεση (residual connection). Η σχέση αυτή προσθέτει την είσοδο x του υποστρώματος στην έξοδο του υποστρώ-

ματος πριν την εφαρμογή της κανονικοποίησης (layer normilization). Το υπόστρωμα μπορεί να είναι είτε προσοχή πολλαπλών κεφαλιών είτε πλήρες συνδεδεμένο στρώμα εμπρόσθιας τροφοδότησης και x η αντίστοιχη είσοδος, ακολουθώντας την εξής μορφή: $LayerNorm(x + Sublayer(x))$. Αυτή η διαδικασία βοηθά στον μετριασμό του προβλήματος της εξαφάνισης κλίσης (vanishing gradient) και επιτρέπει τη διατήρηση της πληροφορίας σε μεγαλύτερο βάθος του δικτύου, καθιστώντας την μάθηση πιο σταθερή και αποτελεσματική.

Αποκωδικοποιητής (Decoder)

Αποτελείται από N πανομοιότυπα στρώματα, το κάθε ένα με τρία κύρια υποστρώματα. Πρώτο υπόστρωμα αποτελεί μια **μάσκα προσοχής πολλαπλών κεφαλιών**. Είναι παρόμοιο με το στρώμα προσοχής πολλαπλών κεφαλιών του κωδικοποιητή, με την βασική διαφορά ότι εδώ είναι μάσκα με σκοπό να αποτρέψει τον αποκωδικοποιητή να εστιάσει σε μελλοντικές θέσεις της ακολουθίας δεδομένων. Η μάσκα προσοχής εξασφαλίζει ότι κάθε θέση στην ακολουθία εξόδου μπορεί να εστιάσει μόνο σε προηγούμενες θέσεις, διατηρώντας έτσι την αυτοαναφορική ιδιότητα του μοντέλου και συνήθως εφαρμόζεται αποδίδοντας την τιμή $-\infty$ στις βαθμολογίες προσοχής των επόμενων μονάδων (tokens) πριν περάσουν από την συνάρτηση ενεργοποίησης softmax. Επόμενο στρώμα αποτελεί η προσοχή κωδικοποιητή-αποκωδικοποιητή. Σε αυτό το στρώμα, οι τιμές των ερωτημάτων Q προέρχονται από το μπλοκ της μάσκας προσοχής του κωδικοποιητή ενώ τα κλειδιά K και οι τιμές V προέρχονται από την έξοδο του τελικού στρώματος του κωδικοποιητή. Αυτό ονομάζεται και **διασταυρούμενη προσοχή** (cross attention) και επιτρέπει στον αποκωδικοποιητή να έχει πρόσβαση στα σχετικά μέρη της κωδικοποιημένης ακολουθίας εισόδου και να αποκτά ολική εποπτεία όταν παράγει το αποτέλεσμα. Τελευταίο στρώμα στην δομή του αποκωδικοποιητή αποτελεί ένα πλήρως συνδεδεμένο στρώμα εμπρόσθιας τροφοδότησης το οποίο είναι παρόμοιο με αυτό του κωδικοποιητή.

Όπως και στον κωδικοποιητή, κάθε υπόστρωμα στον αποκωδικοποιητή περιλαμβάνει μια σχέση παράκαμψης (skip connection) η οποία ακολουθείται από ένα στρώμα κανονικοποίησης. Τέλος, όπως φαίνεται και στο σχήμα 3.5, το Nth στρώμα αποκωδικοποιητή περνά σε ένα άλλο γραμμικό στρώμα, με μέγεθος ίσο με τον αριθμό των κλάσεων που υπάρχουν σε κάθε συγκεκριμένη εργασία. Το γραμμικά μετασχηματισμένο αυτό διάνυσμα, περνά τελικά μέσα από μια συνάρτηση ενεργοποίησης softmax ώστε τελικώς να εξαχθούν πιθανότητες για κάθε κλάση που υπάρχει.

Ενσωματώσεις (Embeddings)

Οι μετασχηματιστές χρησιμοποιούν τις ενσωματώσεις (embeddings) καθώς παίζουν καθοριστικό ρόλο στη μετατροπή των εισερχόμενων διακριτών μονάδων σε πυκνά, συνεχή διανύσματα. Κάθε μονάδα της ακολουθίας εισόδου αντιστοιχεί σε ένα διανυσματικό χώρο σταθερών διαστάσεων. Τα διανύσματα αυτά εκπαιδεύονται κατά τη διάρκεια της διαδικασίας εκμάθησης, επιτρέποντας στο μοντέλο να καταγράφει σημαντικές πληροφορίες και σχέσεις μεταξύ των διαφορετικών τιμών μονάδων (λέξεις, σύμβολα, εικονοστοιχεία κλπ.). Η διαδικασία αυτή εξασφαλίζει ότι μονάδες με παρόμοια σημασία τοποθετούνται κοντά η μια με την άλλη στον ενσωματωμένο χώρο, ενισχύοντας την ικανότητα του μοντέλου να κατανοεί, για παράδειγμα κείμενα και εικόνες, και να αντλεί νοήματα από αυτά.

Κωδικοποιήσεις Σχέσης (Postitional Encodings)

Ο μηχανισμός αυτο-προσοχής δεν λαμβάνει υπόψη τη σειρά των μονάδων όταν επεξεργάζεται διαδοχικά δεδομένα, επειδή δεν βασίζεται σε επαναληπτικές ή συνελικτικές δομές. Αντιθέτως, λειτουργεί παράλληλα και επεξεργάζεται κάθε μονάδα ανεξάρτητα από τη θέση στην ακολουθία. Δεδομένου αυτής της έλλειψης, χρησιμοποιούνται κωδικοποιήσεις θέσης **(positional encodings)**. Οι κωδικοποιήσεις αυτές προστίθενται στα ενσωματωμένα διανύσματα των μονάδων, επιτρέποντας στο μοντέλο να λαμβάνει υπόψη τη θέση κάθε μονάδας μέσα στην ακολουθία. Έχουν την ίδια διάσταση με τις ενσωματώσεις και μπορεί να είναι είτε εκπαιδεύσιμες είτε σταθερές. Οι κωδικοποιήσεις θέσης υπολογίζονται χρησιμοποιώντας συναρτήσεις ημιτόνου και συνημίτονου διαφορετικών συχνοτήτων για κάθε διάσταση του ενσωματωμένου διανύσματος. Συγκεκριμένα, για κάθε θέση pos και διάσταση i, η κωδικοποίηση θέσης υπολογίζεται ως εξής:

$$PE_{(pos,2i)} = \sin\left(\frac{pos}{10000^{\frac{2i}{d_{model}}}}\right) \tag{3.8}$$

$$PE_{(pos,2i+1)} = \cos\left(\frac{pos}{10000^{\frac{2i}{d_{model}}}}\right) \tag{3.9}$$

3.3 Οπτικοί Μετασχηματιστές

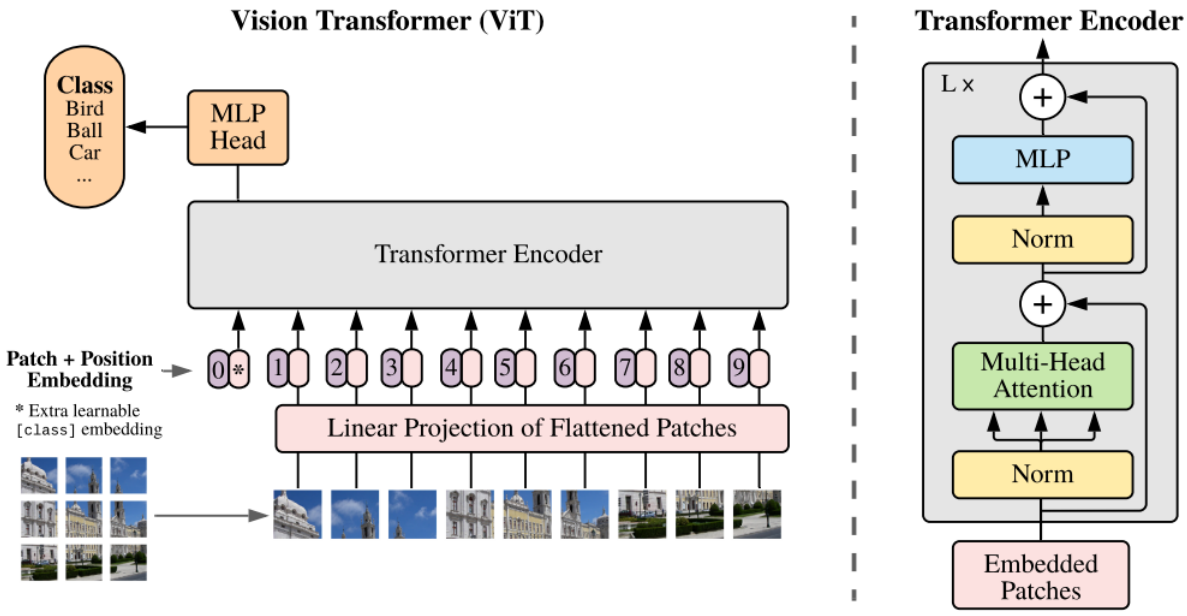
Μέχρι στιγμής, έχει πραγματοποιηθεί ανάλυση της αρχιτεκτονικής των μετασχηματιστών και του μηχανισμού αυτο-προσοχής. Σε επόμενο βήμα, πραγματοποιείται ανάλυση των **οπτικών μετασχηματιστών**, μια από τις πιο δημοφιλείς αρχιτεκτονικές όσον αφορά την επεξεργασία εικόνας σε εφαρμογές όρασης υπολογιστών και αναπτύχθηκε το 2021 [12]. Χρησιμοποιεί μόνο κωδικοποιητή καθώς σκοπός είναι η εξαγωγή σημαντικών χαρακτηριστικών από τις εικόνες και η ταξινόμηση τους. Πρώτα, θα μελετηθεί η διαδικασία μετατροπής της εικόνας σε επιμέρους τμήματα (patches) και η ενσωμάτωση των κωδικοποιήσεων θέσης (positional embeddings) ώστε να διατηρηθεί η χωρικής πληροφορία. Στη συνέχεια, θα αναλυθεί ο τρόπος με τον οποίο οι οπτικοί μετασχηματιστές αξιοποιούν τα στρώματα αυτο-προσοχής και τα νευρωνικά δίκτυα εμπρόσθιας τροφοδότησης (feedforward neural networks) μέσω του κωδικοποιητή για να αναγνωρίσει και να κατανοήσει περίπλοκα μοτίβα και χαρακτηριστικά στην εικόνα. Αμέσως μετά, θα παρουσιαστούν τα πλεονεκτήματα της αρχιτεκτονικής αυτής καθώς και οι προκλήσεις που αντιμετωπίζει όσον αφορά τις απαιτήσεις σε δεδομένα και σε υπολογιστικούς πόρους. Τέλος, θα πραγματοποιηθεί αναφορά σε πολύ σημαντικά για την παρούσα εργασία μοντέλα, τα οποία έχουν αναπτυχθεί με την χρήση οπτικών μετασχηματιστών.

Μετατροπή Εικόνας σε Επιμέρους Τμήματα (Patches)

Πρώτο βήμα για τον οπτικό μετασχηματιστή αποτελεί η διάσπαση της εικόνας σε μικρότερα, σταθερού μεγέθους τμήματα. Μια εικόνα διαστάσεων $H \times W \times C$ (ύψος, πλάτος, κανάλια) διασπάται σε μικρότερα τμήματα σταθερού μεγέθους $P \times P \times C$. Έτσι, για μια εικόνα $H \times W$ με μέγεθος P, ο συνολικός αριθμός των τμημάτων θα είναι $\frac{H}{P} \times \frac{W}{P}$. Κάθε τμήμα επιπεδοποιείται σε ένα μονοδιάστατο διάνυσμα και στη συνέχεια, κάθε διάνυσμα προβάλλεται σε έναν διανυσματικό χώρο χαμηλότερων διαστάσεων μέσω ενός γραμμικού επιπέδου, το οποίο εξάγει τα αρχικά χαρακτηριστικά των τμημάτων. Τα διανύσματα αυτά χαρακτηρίζονται ως ενσωμάτωση τμήματος (*patch embeddings*) και περιέχουν την πληροφορία των οπτικών χαρακτηριστικών κάθε τμήματος.

Ενσωμάτωση των κωδικοποιήσεων θέσης και τμήματος (positional and patch embeddings)

Ένα πολύ σημαντικό χαρακτηριστικό των μοντέλων των μετασχηματιστών είναι η ικανότητα τους να διατηρούν την χωρική πληροφορία της ακολουθίας εισόδου. Στις ακολουθίες εικόνων, αυτό επιτυγχάνεται μέσω της ενσωμάτωσης θέσης και τμήματος (positional and patch embeddings).



Σχήμα 3.6: Επισκόπηση μοντέλου οπτικού μετασχηματιστή. Διαχωρίζεται μια εικόνα σε τμήματα σταθερού μεγέθους, ενσωματώνονται γραμμικά το καθένα από αυτά, προσθέτονται ενσωματώσεις θέσης και τροφοδοτείται η προκύπτουσα ακολουθία διανυσμάτων σε έναν τυπικό κωδικοποιητή μετασχηματιστή. Προκειμένου να πραγματοποιηθεί ταξινόμηση, χρησιμοποιείται η τυπική προσέγγιση της προσθήκης μιας επιπλέον «μονάδας ταξινόμησης» με δυνατότητα εκμάθησης στην ακολουθία - *Πηγή:[12]*

Κάθε τμήμα εικόνας μετατρέπεται σε ένα διάνυσμα χαρακτηριστικών το οποίο ονομάζεται ενσωμάτωση τμήματος, όπως περιγράφηκε στην προηγούμενη παράγραφο. Επειδή τα τμήματα αντιμετωπίζονται ως ακολουθίες και όχι ως εικόνες με χωρική δομή, είναι σημαντικό να ενσωματωθούν πληροφορίες για τη θέση κάθε τμήματος. Αυτό επιτυγχάνεται προσθέτοντας ένα διάνυσμα κωδικοποίησης θέσης σε κάθε ενσωμάτωση τμήματος. Οι ενσωματώσεις θέσης μπορεί να είναι είτε εκπαιδεύσιμες είτε σταθερές, όπως έχει προαναφερθεί στο υποκεφάλαιο 3.2. Τελικό βήμα αποτελεί ο συνδυασμός των ενσωματώσεων τμήματος και θέσης. Αυτό, βοηθά καθοριστικά το μοντέλο να διατηρήσει την χωρική σχέση των τμημάτων.

Ένα πολύ σημαντικό κομμάτι στην αρχιτεκτονική του οπτικού μετασχηματιστή αποτελεί ένα ειδικό μαθησιακό ενσωμάτωμα, γνωστό ως μονάδα ταξινόμησης (classification (CLS) token), το οποίο προστίθεται στην αρχή της ακολουθίας των ενσωματωμένων τμημάτων και θέσεων. Αυτό, χρησιμοποιείται για την εξαγωγή της συνολικής ανα-

παράστασης της εικόνας, η οποία τελικά χρησιμοποιείται για ταξινόμηση ή εξαγωγή σημαντικών χαρακτηριστικών. Η προσθήκη της CLS μονάδας εξασφαλίζει ότι το μοντέλο είναι ικανό να εστιάσει σε σημαντικές χαρακτηριστικές πληροφορίες σε ολόκληρη την εικόνα.

Κωδικοποιητής Μετασχηματιστή και Κεφάλι Πολυεπίπεδου Πλήρους Συνδεδεμένου Δικτύου

Όπως περιγράφεται και στο σχήμα 3.6, οι ενσωματώσεις θέσης και τμήματος περνάνε στον κωδικοποιητή του μετασχηματιστή μοντέλου. Ακολουθεί ακριβώς την ίδια βασική αρχιτεκτονική και διεργασίες όπως περιγράφηκαν στο 3.2 στην ανάλυση του κωδικοποιητή. Τέλος, αφού οι κωδικοποιήσεις έχουν περάσει από τα στρώματα του κωδικοποιητή, η έξοδος της CLS μονάδας χρησιμοποιείται για την τελική ταξινόμηση ή πρόβλεψη μέσω ενός κεφαλιού πολυεπίπεδου πλήρους συνδεδεμένου δικτύου.

Πλεονεκτήματα και Δημοφιλείς Εφαρμογές

Ο οπτικός μετασχηματιστής παρουσιάζει πληθώρα πλεονεκτημάτων τα οποία παρουσιάζονται παρακάτω και πραγματοποιείται επίσης αναφορά σε πολύ δημοφιλή μοντέλα (frameworks) που αξιοποιούν αυτή την αρχιτεκτονική.

Πλεονεκτήματα

- **Κλιμάκωση:** Ο οπτικός μετασχηματιστής μπορεί να χειριστεί πολύ μεγάλα σύνολα δεδομένων και να επωφεληθεί από αυτά κατά την εκπαίδευση του
- **Παράλληλη Επεξεργασία:** Γενικότερα οι μετασχηματιστές, επιτρέπουν την παράλληλη επεξεργασία κατά την εκπαίδευση, οδηγώντας και σε ταχύτερους χρόνους εκπαίδευσης. Αυτό σημαίνει ότι ο οπτικός μετασχηματιστής μπορεί να επεξεργάζεται πολλαπλά στοιχεία των δεδομένων ταυτόχρονα, εκμεταλλεύοντας τον μηχανισμό αυτο-προσοχής που επιτρέπει την ανεξάρτητη επεξεργασία κάθε στοιχείου της ακολουθίας. Η παράλληλη επεξεργασία βελτιώνει την αποδοτικότητα, επιτρέπει ταχύτερη εκπαίδευση και πραγματοποιεί αποδοτική χρήση των υπολογιστικών πόρων όπως Μονάδες Επεξεργασίας Γραφικών(GPUs[1]).
- **Απόδοση:** Σε μεγάλα σύνολα δεδομένων, ο οπτικός μετασχηματιστής έχει δείξει ότι υπερέχει έναντι των παραδοσιακών συνελικτικών νευρωνικών δικτύων (CNNs[2])

[1]Graphics Processing Units
[2]Convolutional Neural Networks

Δημοφιλείς Εφαρμογές

- **MAE (Masked Autoencoders):** Χρησιμοποιεί μάσκες τμημάτων για αυτο-επιβλεπόμενη μάθηση.
- **DINO (Self-Distillation with No Labels):** Εφαρμόζει μια προσέγγιση αυτο-επιβλεπόμενης μάθησης.
- **CLIP (Contrastive Language-Image Pre-Training):** Εκπαιδεύεται σε μεγάλης κλίμακας ζεύγη εικόνας και κειμένου για εκμάθηση μεταφερόμενων οπτικών μοντέλων από εποπτεία φυσικής γλώσσας.
- **JEPA (Joint Embedding Predictive Architecture):** Επικεντρώνεται σε κοινή ενσωμάτωση για βελτιωμένη προβλεπτική μοντελοποίηση.

Τα παραπάνω έργα θα αναλυθούν λεπτομερώς στο κεφάλαιο 3, καθώς αποτελούν σχετικές εφαρμογές με την παρούσα εργασία.

ΚΕΦΑΛΑΙΟ 4

Σχετικές Εργασίες στη Βιβλιογραφία

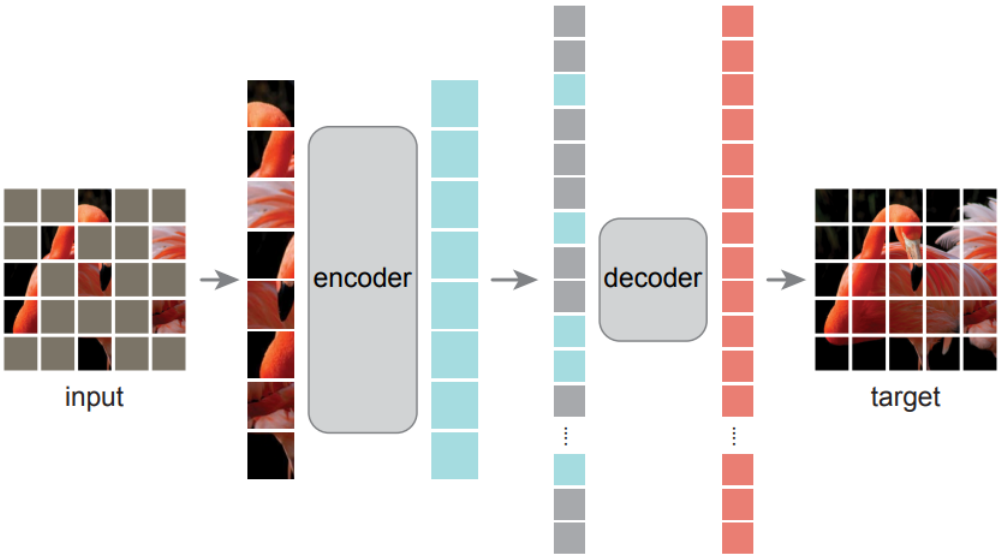
Στο κεφάλαιο αυτό περιγράφονται οι βασικές σχετικές εφαρμογές που έχουν υλοποιηθεί στην βιβλιογραφία και θα αποτελέσουν σημαντικά έργα για την εκπόνηση της παρούσας εργασίας.

4.1 Masked Autoencoders Are Scalable Vision Learners (MAE)

Η ερευνητική εφαρμογή ”Masked Autoencoders Are Scalable Vision Learners” [15] εισάγει μια νέα προσέγγιση στην αυτο-επιβλεπόμενη μάθηση στην όραση υπολογιστών, εστιάζοντας στους αυτο-κωδικοποιητές με χρήση μασκών (Masked Autoencoders). Ένα αυτο-επιβλεπόμενο σύστημα παίρνει την είσοδο (εικόνα) και την ανακατασκευάζει, χωρίς να χρειάζεται επισημασμένα δεδομένα, μαθαίνοντας γενικές αναπαραστάσεις της μορφής που έχει η είσοδος. Η αρχιτεκτονική MAE χρησιμοποιεί αυτή την ιδιότητα των αυτο-κωδικοποιητών και εισάγει και την έννοια της μάσκας της εικόνας. Η ερευνητική εξήγηση πίσω από αυτή την ιδέα είναι ότι θα αφαιρεθούν εικονοστοιχεία από την εικόνα και το μοντέλο θα τροφοδοτηθεί με ελλιπή είσοδο και στόχος είναι να μάθει πως έμοιαζε η πλήρης αρχική εικόνα. Αυτή η τεχνική επιτρέπει στο μοντέλο να αναπτύξει μια πιο ισχυρή κατανόηση του περιεχομένου της εικόνας, καθώς πρέπει να συμπεράνει τα λείποντα μέρη με βάση το ορατό πλαίσιο.

Ένα σημαντικό στοιχείο της αρχιτεκτονικής MAE είναι η εφαρμογή της στη γραμμική χρήση χαρακτηριστικών (**linear probing**). Η γραμμική χρήση περιλαμβάνει τη χρήση των αναπαραστάσεων που μαθαίνονται από το MAE για μεταγενέστερες εργασίες με έναν απλό γραμμικό ταξινομητή, αξιολογώντας αποτελεσματικά την ποιότητα αυτών των αναπαραστάσεων. Οι συγγραφείς χρησιμοποιούν τη γραμμική χρήση για να αξιολογήσουν πόσο καλά μπορεί να αποτυπώσει χρήσιμα χαρακτηριστικά από τις εικόνες. Επιπρόσθετα, δείχνουν ότι οι αναπαραστάσεις που μαθαίνονται από το MAE, όταν χρησιμοποιούνται σε εργασίες γραμμικής χρήσης χαρακτηριστικών, ξεπερνούν

αυτές που μαθαίνονται από άλλες μεθόδους αυτο-επιβλεπόμενης και επιβλεπόμενης μάθησης. Αυτό είναι ιδιαίτερα εντυπωσιακό γιατί δείχνει ότι μπορεί να μάθει εξαιρετικά αποτελεσματικά χαρακτηριστικά με μια σχετικά απλή αρχιτεκτονική και χωρίς την ανάγκη για μεγάλα σύνολα επισημασμένων δεδομένων. Τα αποτελέσματα της γραμμικής χρήσης επικυρώνουν την ικανότητα του MAE να γενικεύει και να μεταφέρει τα εκπαιδευμένα χαρακτηριστικά σε διάφορες κατάντη εργασίες (downstream tasks).



Σχήμα 4.1: Αρχιτεκτονική MAE. Κατά την προ-εκπαίδευση, ένα μεγάλο τυχαίο υποσύνολο τμημάτων εικόνας (π.χ. 75%) καλύπτεται. Ο κωδικοποιητής εφαρμόζεται στο μικρό υποσύνολο των ορατών τμημάτων. Οι μονάδες μάσκας εισάγονται μετά τον κωδικοποιητή και το πλήρες σύνολο των κωδικοποιημένων τμημάτων και των μονάδων μάσκας υποβάλλεται σε επεξεργασία από έναν μικρό αποκωδικοποιητή που αναδομεί την αρχική εικόνα σε εικονοστοιχεία. Μετά την προ-εκπαίδευση, ο αποκωδικοποιητής απορρίπτεται και ο κωδικοποιητής εφαρμόζεται σε μη αλλοιωμένες εικόνες (πλήρη σετ τμημάτων) για εργασίες αναγνώρισης - *Πηγή:[15]*

Στο σχήμα 4.1 απεικονίζεται η αρχιτεκτονική του MAE. Κατά τη διάρκεια της προεκπαίδευσης, ένα μεγάλο υποσύνολο των τμημάτων της εικόνας εισόδου επικαλύπτεται (75%). Στην είσοδο του κωδικοποιητή εισέρχεται το μικρό υποσύνολο των ορατών τμημάτων της εικόνας (25%). Τα επικαλυπτόμενα τμήματα εισάγονται μετά τον κωδικοποιητή και το σύνολο των επικαλυπτόμενων και κωδικοποιημένων τμημάτων (patches) εισέρχονται και επεξεργάζονται στον αποκωδικοποιητή, ο οποίος ανακατασκευάζει την αρχική εικόνα σε εικονοστοιχεία. Μετά την προεκπαίδευση, ο αποκωδικοποιητής απορρίπτεται και εφαρμόζεται ο κωδικοποιητής στα τμήματα ολόκληρων εικόνων για

εργασίες αναγνώρισης.

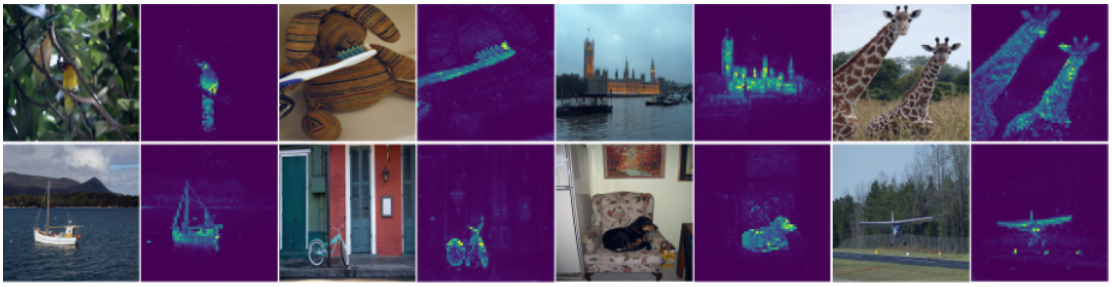
4.2 Emerging Properties in Self-Supervised Vision Transformers (DINO)

Το DINO (Distillation with No Labels) [11] είναι μια μέθοδος αυτο-επιβλεπόμενης μάθησης προσαρμοσμένη στην αρχιτεκτονική του οπτικού μετασχηματιστή [12]. Χρησιμοποιεί ένα μοντέλο (framework) δασκάλου-μαθητή (teacher-student), όπου οι παράμετροι του δικτύου του δασκάλου ενημερώνονται μέσω ενός εκθετικού κινούμενου μέσου των παραμέτρων του δικτύου του μαθητή. Αυτή η μέθοδος ονομάζεται αυτόδιάχυση (**self-distillation**) και είναι ένα είδος διάχυσης γνώσης όπου ένα μεμονωμένο νευρωνικό δίκτυο (ο μαθητής) μαθαίνει από τις δικές του προβλέψεις που έγιναν κατά τη διάρκεια προηγούμενων φάσεων εκπαίδευσης. Εδώ, αυτή διαδικασία επιτρέπει στο δίκτυο του μαθητή να μάθει σημαντικά χαρακτηριστικά χωρίς επισημασμένα δεδομένα (labeled data), συγκρίνοντας τις εξόδους του με αυτές του δικτύου δασκάλου. Μια βασική πτυχή της μεθόδου DINO είναι ότι χρησιμοποιεί πολλαπλές όψεις μιας εικόνας σε διαφορετικές αναλύσεις (**multi-crop augmentation**) για το στάδιο της εκπαίδευσης, ενισχύοντας την ικανότητα γενίκευσης.

Η μέθοδος αυτή είναι σχεδιασμένη για να χρησιμοποιεί την δομή του Vision Transformer και του μηχανισμού προσοχής για να καταγράψει αποτελεσματικά γενική και τοπική πληροφορία (global-local). Το DINO χρησιμοποιεί την απώλεια διασταυρούμενης εντροπίας (cross-entropy loss) για να ευθυγραμμιστούν οι έξοδοι του δικτύου μαθητή με αυτές του δασκάλου, διασφαλίζοντας ότι τα χαρακτηριστικά που μαθαίνονται είναι συνεπή και ανθεκτικά σε διαφορετικές προσαρμογές. Όλα τα παραπάνω, επιτρέπουν στην μέθοδο αυτή να επιτυγχάνει υψηλές επιδόσεις στην αυτο-επιβλεπόμενη μάθηση, με σημαντικές βελτιώσεις σε σχέση με προηγούμενες μεθόδους όσον αφορά εφαρμογές όπως ταξινόμηση εικόνων και ανίχνευση αντικειμένων.

Ένα σημαντικό χαρακτηριστικό στην μέθοδο DINO είναι η χρήση της γραμμικής χρήσης (**linear probing**), η οποία αξιολογεί την ποιότητα των χαρακτηριστικών που μαθαίνονται μέσω της εκπαίδευσης ενός γραμμικού ταξινομητή πάνω στα παγωμένα χαρακτηριστικά που εξάγονται από το προ-εκπαιδευμένο δίκτυο. Τα χαρακτηριστικά του DINO, έχουν αποδειχθεί ιδιαίτερα αποτελεσματικά, επιτυγχάνοντας υψηλή ακρίβεια σε δημοφιλή σετ δεδομένων και σημεία αναφοράς (benchmarks) όπως το ImageNet. Αυτό αποδεικνύει ότι τα χαρακτηριστικά που μαθαίνονται μέσω του DINO δεν είναι μόνο ισχυρά

αλλά και μεταβιβάσιμα, καθιστώντας τα πολύτιμα για κατάντη εργασίες (downstream tasks) σε εφαρμογές όρασης υπολογιστών.



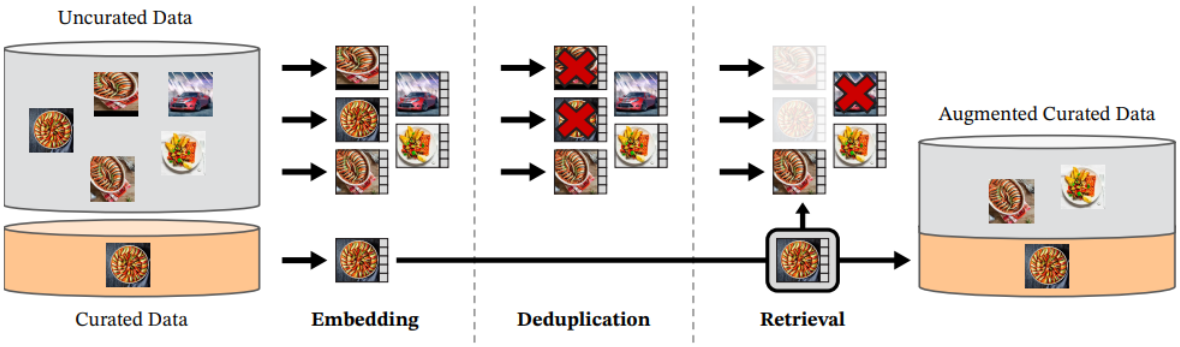
Σχήμα 4.2: Αυτο-προσοχή από ένα οπτικό μετασχηματιστή με 8×8 τμήματα, εκπαιδευμένο χωρίς επίβλεψη. Εξετάζεται η αυτο-προσοχή της μονάδας ταξινόμησης [CLS] στα κεφάλια του τελευταίου στρώματος. Αυτή η μονάδα δεν συνδέεται με καμία ετικέτα ή επίβλεψη. Αυτοί οι χάρτες δείχνουν ότι το μοντέλο μαθαίνει αυτόματα χαρακτηριστικά για συγκεκριμένη κατηγορία που οδηγούν σε τμηματοποιήσεις αντικειμένων χωρίς επίβλεψη -*Πηγή:[19]*

4.3 DINOv2: Learning Robust Visual Features without Supervision

Η μέθοδος DINOv2 [19], είναι μια επέκταση της μεθόδου DINO [11] που κλιμακώνει το μοντέλο και το μέγεθος του σετ δεδομένων, εισάγοντας τεχνικές βελτιώσεις για την επιτάχυνση και σταθεροποίηση της εκπαίδευσης. Το DINOv2 χρησιμοποιεί έναν οπτικό μετασχηματιστή [12] με 1 δισεκατομμύριο παραμέτρους και ένα επιμελημένο σετ δεδομένων 142 εκατομμυρίων εικόνων (σχήμα 4.3).
Στόχος του DINOv2 είναι να βελτιώσει την αποτελεσματικότητα των αυτο-επιβλεπόμενων μεθόδων και το επιτυγχάνει εφαρμόζοντας καινοτομίες όπως πολλαπλές περικοπές κατά την εκπαίδευση χρησιμοποιώντας FlashAttention[1] και έναν κανονικοποιητή για την ομοιόμορφη διάδοση των χαρακτηριστικών. Η μέθοδος αυτή χρησιμοποιεί επίσης έναν συνδυασμό απωλειών DINO [11] και iBOT [18] με ξεχωριστά κεφάλια προβολής με δυνατότητα εκμάθησης για καλύτερη απόδοση σε κλίμακα.

Μια αξιοσημείωτη διαφορά μεταξύ των μοντέλων DINO και DINOv2 είναι η έμφαση του δεύτερου στην επεκτασιμότητα και την αποτελεσματικότητα. Αξιοποιεί μεγαλύτερα σύνολα δεδομένων και μεγέθη μοντέλων, ενσωματώνοντας προηγμένες τεχνικές

[1] Αποτελεσματικός αλγόριθμος που έχει σχεδιαστεί για να επιταχύνει τον μηχανισμό προσοχής στους μετασχηματιστές μειώνοντας παράλληλα τη χρήση μνήμης



Σχήμα 4.3: Επισκόπηση του αγωγού επεξεργασίας δεδομένων. Οι εικόνες από επιμελημένες και μη επιμελημένες πηγές δεδομένων αντιστοιχίζονται πρώτα σε ενσωματώσεις. Στη συνέχεια, οι μη επεξεργασμένες εικόνες αφαιρούνται από το αντίγραφο πριν αντιστοιχιστούν με επεξεργασμένες εικόνες. Ο συνδυασμός που προκύπτει αυξάνει το αρχικό σύνολο δεδομένων μέσω ενός αυτο-επιβλεπόμενου συστήματος ανάκτησης - *Πηγή:[19]*

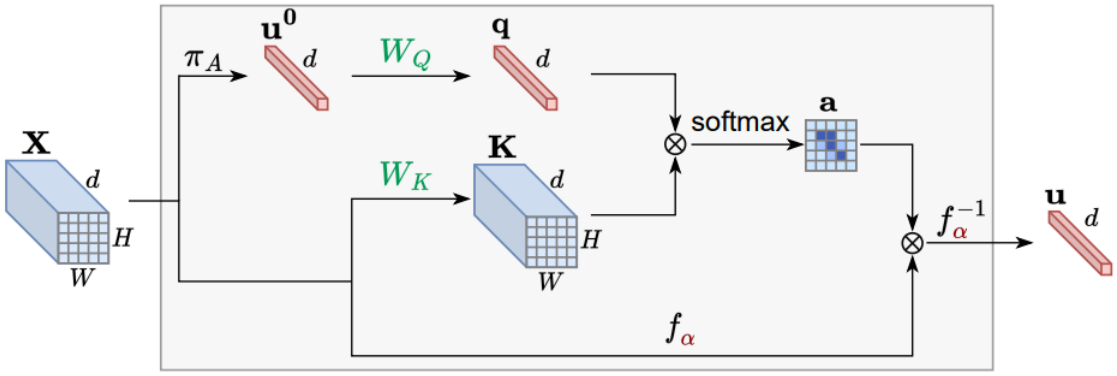
όπως η ομαδοποίηση ακολουθιών (sequence packing) και ο πλήρως μοιρασμένος παραλληλισμός δεδομένων για τη διαχείριση του αυξημένου υπολογιστικού φόρτου. Όλες οι παραπάνω βελτιώσεις επιτρέπουν στο DINOv2 να πετυχαίνει υψηλότερες ακρίβειες και πιο ισχυρή εκμάθηση χαρακτηριστικών, γεγονός το οποίο επιβεβαιώνεται από την μεγαλύτερη απόδοση που έχει αποδείξει σε διάφορες εργασίες αναφοράς (benchmarks), συμπεριλαμβανομένου των εργασιών γραμμικής χρήσης χαρακτηριστικών (linear probing), όπου ξεπερνά προηγούμενα μοντέλα αιχμής (MAE [15], DINO [11])

4.4 Keep It SimPool: Who Said Supervised Transformers Suffer from Attention Deficit?

Η ερευνητική εφαρμογή **SimPool** [20], προτείνει την αντικατάσταση του προεπιλεγμένου μηχανισμού προσοχής στον κωδικοποιητή είτε των οπτικών μετασχηματιστών είτε των συνελικτικών δικτύων, με έναν απλό μηχανισμό συγκέντρωσης βασισμένο στον μηχανισμό προσοχής (attention-based pooling mechanism). Ο μηχανισμός SimPool επιδιώκει να βελτιώσει την ποιότητα των χαρτών προσοχής (Attention Maps) που παράγονται από τα επιβλεπόμενα μοντέλα, φτάνοντας σε επίπεδα αντίστοιχα με αυτά των αυτο-επιβλεπόμενων μεθόδων. Είναι μια μέθοδος σχεδιασμένη να ενσωματώνεται εύκολα και να βελτιώνει την απόδοση των μοντέλων σε διάφορες εφαρμογές όρασης

υπολογιστών, διατηρώντας τα σημαντικά χαρακτηριστικά των δεδομένων.

Η καινοτομία του SimPool έγκειται στη χρήση διασταυρούμενης προσοχής (cross attention), η οποία επιτρέπει την ενσωμάτωση πληροφοριών από διαφορετικές θέσεις του εισερχόμενου σήματος στο μοντέλο. Αυτή η διαδικασία ενισχύει την ικανότητα του κατανοεί τις σχέσεις μεταξύ των απομακρυσμένων σημείων της εικόνας, παράγοντας έτσι αναπαραστάσεις που αποτυπώνουν με μεγαλύτερη ακρίβεια τα όρια των αντικειμένων. Η προσέγγιση αυτή αποδεικνύεται ιδιαίτερα αποτελεσματική, καθώς βελτιώνει την απόδοση των μοντέλων σε προκαταρκτικές εκπαιδευτικές και κατάντη εργασίες (pre-training-downstream tasks), ανεξαρτήτως του τύπου της αρχιτεκτονικής που χρησιμοποιείται.



Σχήμα 4.4: Επισκόπηση του SimPool. Δεδομένου ενός εισερχόμενου τανυστή $X \in \mathbb{R}^{d\times W\times H}$ που γίνεται επίπεδος σε $X \in \mathbb{R}^{d\times p}$ με $p := W \times H$ patches, μία ροή σχηματίζει την αρχική αναπαράσταση $u_0 = \pi_A(X) \in \mathbb{R}^d$ (12) μέσω της ολικής μέσης συγκέντρωσης (GAP), που χαρτογραφείται από το $W_Q \in \mathbb{R}^{d\times d}$ (13) για να σχηματίσει το διανύσμα ερωτήματος $q \in \mathbb{R}^d$. Μία άλλη ροή χαρτογραφεί το X από το $W_K \in \mathbb{R}^{d\times d}$ (14) για να σχηματίσει το κλειδί $K \in \mathbb{R}^{d\times p}$, το οποίο παρουσιάζεται ως τανυστής K. Στη συνέχεια, το q και το K αλληλεπιδρούν για να δημιουργήσουν τον χάρτη προσοχής $a \in \mathbb{R}^p$ (15). Τέλος, η συγκεντρωμένη αναπαράσταση $u \in \mathbb{R}^d$ είναι μια γενικευμένη σταθμισμένη μέση τιμή του X με το a να καθορίζει τα βάρη και η βαθμωτή συνάρτηση f_α να καθορίζει τη λειτουργία συγκέντρωσης - *Πηγή:[20]*

Στο σχήμα 4.4 απεικονίζεται η αρχιτεκτονική του μηχανισμού SimPool. Παρακάτω, ακολουθεί μια περιγραφή των βασικών στοιχείων και βημάτων του μηχανισμού όπως φαίνονται στην εικόνα:

1. **Είσοδος X:** Ο αρχικός τένσορας εισόδου $X \in \mathbb{R}^{d\times W\times H}$ (με διαστάσεις πλάτους W και ύψους H) διανυσματοποιείται σε $X \in \mathbb{R}^{d\times p}$, όπου $p = W \times H$ τμήματα

2. **Αρχική Αναπαράσταση** u^0: Μια ροή δεδομένων δημιουργεί την αρχική αναπαράσταση $u^0 = \pi_A(X) \in \mathbb{R}^d$ μέσω του ολικού μέσου όρου (Global Average Pooling (GAP)), που χαρτογραφείται από $W_Q \in \mathbb{R}^{d \times d}$ για να σχηματίσει το διάνυσμα ερωτήματος $q \in \mathbb{R}^d$

3. **Κλειδί** K**:** Μια άλλη ροή δεδομένων χαρτογραφεί το X μέσω $W_K \in \mathbb{R}^{d \times d}$ για να σχηματίσει το κλειδί $K \in \mathbb{R}^{d \times p}$, που παρουσιάζεται ως τένσορας K

4. **Χάρτης Προσοχής** a**:** Το q και K αλληλεπιδρούν για να δημιουργήσουν τον χάρτη προσοχής $a \in \mathbb{R}^p$

5. **Τελική αναπαράσταση** u**:** Η τελική αναπαράσταση $u \in \mathbb{R}^d$ είναι ένας γενικευμένος σταθμισμένος μέσος όρος του X με το a να καθορίζει τα βάρη και τη συνάρτηση f_α να καθορίζει τη λειτουργία της συλλογής.

4.5 Τεχνικές Προσεκτικής Χρήσης

Στην βιβλιογραφία υπάρχουν πολλές σύγχρονες μέθοδοι (state-of-the-art) οι οποίες αποσκοπούν στην βελτίωση της ολικής αναπαράστασης αλλά και γενικότερα των εφαρμογών στο πεδίο της όρασης υπολογιστών. Ορισμένες μέθοδοι, ακόμα και αν δεν έχουν χρησιμοποιηθεί για τεχνικές γραμμικής και προσεκτικής χρήσης ή ακόμα και για τεχνικές συγκέντρωσης (pooling methods), στα πλαίσια της παρούσας διπλωματικής εργασίας τις εφαρμόσαμε σαν τέτοιες τεχνικές ώστε να γίνουν δίκαιες συγκρίσεις και να εξαχθούν έγκυρα συμπεράσματα. Παρακάτω, πραγματοποιείται μια πρώτη ανάλυση και σύγκριση των αρχιτεκτονικών της κάθε μεθόδου (Πίνακας 4.1).

Συγκεκριμένα, οι μέθοδοι **SigLIP** [21], **V-Jepa** [22] και **CaiT** [14] έχουν παρόμοιες αρχιτεκτονικές για την μη γραμμική χρήση με μηχανισμό προσοχής, αρχικοποιούν τις παραμέτρους των ερωτημάτων (queries) με εκπαιδεύσιμα διανύσματα και χρησιμοποιούν γραμμικούς μετασχηματισμούς τόσο στα ερωτήματα (queries), όσο και στα κλειδιά (keys) και τις τιμές (values). Επιπλέον, χρησιμοποιούν έναν γραμμικό μετασχηματισμό στις εξόδους του διασταυρούμενου μηχανισμού προσοχής (cross attention). Τέλος, η αρχιτεκτονική τους επεκτείνεται περαιτέρω με την προσθήκη ενός πλήρους συνδεδεμένου δικτύου (MLP) το οποίο αποτελείται από δύο γραμμικούς μετασχηματισμούς και μια σχέση μη γραμμικότητας ενδιάμεσα τους (ReLU). Η βασική διαφορά στην αρχιτεκτονική τους έγκειται στο ότι το V-Jepa χρησιμοποιεί κανονικοποίηση στρώματος στα κλειδιά και στις τιμές, το CaiT στα ερωτήματα, τα κλειδιά και τις τιμές και το SigLIP δεν

Pooling Method	ViT-Small Params	ViT-Base Params	Initialization	Layer Norm	Pos. Embed	Linears	# Heads	Final Linear	Extra MLP
Simpool	681448	1951720	GAP	K, V		Q,K	1		
Simpool w/out linears	386536	772072	GAP	Q, K		-	1		
CLIP	1075816	3330280	GAP	-	✓	Q,K,V	4	✓	
SigLIP	2159080	7856104	Learnable	-		Q,K,V	8	✓	✓
V-Jepa	2160616	7859176	Learnable	K, V		Q,K,V	12	✓	✓
AIM	681064	1950952	Learnable	K,V (BN)		K,V	12		
CaiT	2162152	7862248	Learnable	Q, K, V		Q,K,V	8	✓	✓
CoCa	927208	1853416	Learnable	Q		Q,K,V	8	✓	
ViT Block	2159080	7856104	P.Tokens	Q,K,V		Q,K,V	8		✓
CBAM	404300	844364	Channel Attn: 2 - 1×1 Convolutions, Spatial Attn: 1 - 7×7 Convolution						

Πίνακας 4.1: Σύγκριση χαρακτηριστικών και υπολογιστικού κόστους των αρχιτεκτονικών διαφόρων μεθόδων που χρησιμοποιήθηκαν ως μέθοδοι προσεκτικής χρήσης και συγκέντρωσης.

χρησιμοποιεί κάποια τέτοια τεχνική. Λόγω τη χρήσης του γραμμικού μετασχηματιστή και του MLP δικτύου, το βάθος των μοντέλων είναι μεγάλο και απαιτούν πολλές παραμέτρους, όπως φαίνεται και στον Πίνακα 4.1. Η μέθοδος **AIM** [23], αρχικοποιεί τις παραμέτρους των ερωτημάτων με εκπαιδεύσιμα διανύσματα, χρησιμοποιεί κανονικοποίηση παρτίδας σε κλειδιά και τιμές και γραμμικούς μετασχηματιστές τόσο στα κλειδιά όσο και στις τιμές. Η μέθοδος αυτή, δεν χρησιμοποιεί κάποιο γραμμικό μετασχηματισμό στις εξόδους του διασταυρούμενου μηχανισμού προσοχής καθώς ούτε και κάποιο επιπλέον πλήρες συνδεδεμένο δίκτυο με αποτέλεσμα οι παράμετροι που απαιτούνται να είναι πάρα πολύ χαμηλοί σε αριθμό. Αντιθέτως, η μέθοδος **CoCa** [17] χρησιμοποιεί γραμμικούς μετασχηματιστές στα ερωτήματα, κλειδιά και τιμές και χρησιμοποιεί κανονικοποίηση στρώματος στα ερωτήματα, αρχικοποιώντας τις παραμέτρους τους με εκπαιδεύσιμα διανύσματα. Χρησιμοποιεί επίσης γραμμικό μετασχηματισμό στις εξόδους του διασταυρούμενου μηχανισμού προσοχής αλλά όχι κάποιο MLP δίκτυο και η σχετικά πιο απλή αρχιτεκτονική του αντικατοπτρίζεται στον αριθμό παραμέτρων που απαιτεί, οποιίος είναι σχετικά χαμηλός. Ακολούθως, η μέθοδος ViT Block, όπως έχει υλοποιηθεί από τον οπτικό μετασχηματισή, διαφέρει αρκετά από τις υπόλοιπες τεχνικές καθώς χρησιμοποιεί αυτο-προσοχή και όχι διασταυρούμενη προσοχή και η αρχικοποίηση πραγματοποιείται χρησιμοποιώντας την είσοδο (x) δηλαδή τις μονάδες τμημάτων (patch tokens) και όχι κάποιο εκπαιδεύσιμο διάνυσμα ή GAP για την αρχικοποίηση των ερωτημάτων, κλειδιών και τιμών. Σε αυτά πραγματοποιείται κανονικοποίηση στρώματος και χρησιμοποιείται ένα MLP δίκτυο και τελικώς παρατηρείται ότι το υπολογιστικό κόστος είναι αρκετά μεγάλο και φτάνει το μέγεθος των παραμέτρων που έχει το SigLIP και το V-Jepa.

Οι μέθοδοι **Simpool** [20], Simpool χωρίς γραμμικά επίπεδα **(Simpool w/out linear)** και **CLIP** [13] αρχικοποιούν τις παραμέτρους των ερωτημάτων με τη χρήση ολικού μέσου όρου (GAP) το οποίο δεν είναι εκπαιδεύσιμο, όπως τις άλλες μεθόδους. Αυτό είναι και το μοναδικό κοινό τους στοιχείο αφού παρουσιάζουν αρκετές διαφορές μεταξύ τους στα υπόλοιπα χαρακτηριστικά. Το Simpool πραγματοποιεί κανονικοποίηση στρώματος σε κλειδιά και τιμές και εισάγει γραμμικούς μετασχηματισμούς στα ερωτήματα και στα κλειδιά ενώ η μέθοδος Simpool w/out linears πραγματοποιεί μόνο κανονικοποίηση στρώματος σε ερωτήματα και κλειδιά. Οι δύο αυτές μέθοδοι χαρακτηρίζονται για την απλότητα της αρχιτεκτονικής τους και το χαμηλό υπολογιστικό τους κόστος. Το CLIP από την άλλη μεριά δεν χρησιμοποιεί κανονικοποίηση στρώματος αλλά κωδικοποίηση θέσης. Εισάγει επίσης γραμμικούς μετασχηματισμούς στα ερωτήματα, στα κλειδιά και στις τιμές και ένα γραμμικό μετασχηματισμό στις εξόδους του διασταυρούμενου μηχανισμού προσοχής. Η απαίτηση αυτής της μεθόδου σε παραμέτρους μπορεί να χαρακτηριστεί σχετικά υψηλή. Τέλος, η μέθοδος **CBAM** [9] χρησιμοποιεί μια τελείως διαφορετική αρχιτεκτονική με όλες τις παραπάνω μεθόδους, καθώς χρησιμοποιεί μηχανισμούς συνελικτικής προσοχής (convolutional attention mechanisms) αντί για κεφάλια προσοχής των μετασχηματιστών, καθιστώντας τη μοναδική σε αυτή τη σύγκριση. Εισάγει ένα κανάλι προσοχής το οποίο εστιάζει στο ”τι” είναι σημαντικό στα χαρακτηριστικά των δεδομένων εισόδου και μια χωρική προσοχή η οποία εστιάζει στο ”που”, δηλαδή σε ποια θέση είναι το σημαντικό χαρακτηριστικό αντλώντας της πληροφορία της τοποθεσίας. Η μέθοδος αυτή χαρακτηρίζεται για το χαμηλό υπολογιστικό της κόστος καθώς η απαίτηση σε παραμέτρους είναι χαμηλή.

ΚΕΦΑΛΑΙΟ 5

Πειραματικές Μέθοδοι και Υλοποίηση

Η μέθοδος που προτείνεται και ερευνάται στην παρούσα εργασία ονομάζεται προσεκτική χρήση χαρακτηριστικών με αξιοποίηση του μηχανισμού προσοχής "**Attentive Probing**", χρησιμοποιώντας τη μέθοδο συγκέντρωσης Simpool [20]. Η μέθοδος αυτή χρησιμοποιείται σε κατάντη εργασίες (downstream tasks) γνωστών μεθόδων (frameworks) που έχουν προαναφερθεί, εισάγοντας μια διασταυρούμενη προσοχή (cross attention) στη διαδικασία αξιολόγησης (evaluation process). Σε αυτή, το ερώτημα (query) αρχικοποιείται ως ένας μετασχηματισμός του ολικού μέσου όρου (global average pooling (GAP)), ενώ τα κλειδιά (Q) και οι τιμές (V) είναι τα ίδια τα χαρακτηριστικά. Αυτή η διάταξη επιτρέπει στο μοντέλο να χρησιμοποιεί μηχανισμούς προσοχής για να βελτιώνει την γενική αναπαράσταση (global representation) που εξάγεται από τα χαρακτηριστικά. Οι γραμμικές μετατροπές για το Q και V διασφαλίζουν ότι ο μηχανισμός προσοχής μπορεί να συλλάβει αποτελεσματικά σχετικές αλληλεπιδράσεις μεταξύ των χαρακτηριστικών. Παρά την εισαγωγή αυτών των επιπρόσθετων παραμέτρων, ο αριθμός τους είναι ελάχιστος σε σχέση με τον συνολικό αριθμό παραμέτρων του προεκπαιδευμένου μοντέλου. Τα προ-εκπαιδευμένα μοντέλα στα οποία θα εφαρμοστεί αυτή η μέθοδος είναι τα MAE [15], DINO [11] και DINOv2 [19] και τα τρία βασισμένα στην αρχιτεκτονική του οπτικού μετασχηματιστή.

5.1 Δεδομένα

Το σετ δεδομένων βάση του οποίου υλοποιούνται όλες οι πειραματικές μέθοδοι είναι το δημοφιλές **ImageNet** [1]. Το ImageNet είναι ένα μεγάλο και επισημασμένο σύνολο δεδομένων εικόνων και έχει σχεδιαστεί για να διευκολύνει την έρευνα σε εφαρμογές οπτικής αναγνώρισης. Αυτό το σύνολο δεδομένων αποτελεί τη βάση για πολλές εργασίες ταξινόμησης εικόνων (classification tasks), ανίχνευσης αντικειμένων (object detection) και τμηματοποίησης (segmentation tasks), προσφέροντας ένα σημείο ανα-

φοράς (benchmark) για την αξιολόγηση της απόδοσης διαφόρων αλγορίθμων.

Σύνθεση των Δεδομένων

Το ImageNet αποτελείται από περισσότερες από 14 εκατομμύρια εικόνες κατανεμημένες σε πάνω από 20.000 κατηγορίες. Κάθε εικόνα είναι επισημασμένη με έναν ή περισσότερους περιγραφικούς όρους (synsets) από την ιεραρχία του WordNet[1]. Αυτοί οι όροι είναι οργανωμένοι σε δομή δέντρου, επιτρέποντας την εξερεύνηση τόσο λεπτομερών όσο και γενικών κατηγοριών. Το σύνολο δεδομένων περιλαμβάνει πληθώρα αντικειμένων, ζώων, σκηνών και γεγονότων παρέχοντας αντιπροσωπευτικά δείγματα πραγματικών εικόνων.

Συλλογή Δεδομένων

Οι εικόνες του ImageNet έχουν συλλεχθεί μέσω ενός συνδυασμού διαδικτυακών αναζητήσεων και συνεισφορών χρηστών. Με σκοπό να πραγματοποιηθεί επαλήθευση της ποιότητας των επισημάνσεων (ετικετών), οι δημιουργοί χρησιμοποίησαν πλατφόρμες πληθοπορισμού (crowdsourcing) όπου σχολιαστές επαλήθευσαν και βελτίωσαν τις ετικέτες. Με αυτόν τον σχολαστικό τρόπο επισήμανσης, διασφαλίστηκε ότι οι εικόνες επισημαίνονται με ακρίβεια, ενισχύοντας την αξιοπιστία του συνόλου δεδομένων για εκπαίδευση στην δοκιμή μοντέλων μηχανικής μάθησης.

Διαγωνισμοί ImageNet

Ο διαγωνισμός ImageNet Large Scale Visual Recognition Challenge (ILSVRC) είναι ένας ετήσιος διαγωνισμός που έχει οδηγήσει καθοριστικά στην πρόοδο των εφαρμογών στην όραση υπολογιστών. Το ILSVRC χρησιμοποιεί ένα υποσύνολο του συνόλου δεδομένων που περιλαμβάνει 1000 κατηγορίες και περίπου 1,2 εκατομμύρια εικόνες εκπαίδευσης (train set), 50.000 εικόνες επαλήθευσης (validation set) και 100.000 εικόνες για δοκιμή (test set). Η παρούσα εργασία χρησιμοποιεί αυτό το υποσύνολο για τις πειραματικές μεθόδους και υλοποιήσεις που θα αναλυθούν διεξοδικά παρακάτω.

5.2 Πειραματική Διαδικασία

Όλα τα πειράματα της παρούσας εργασίας πραγματοποιήθηκαν σε δύο κάρτες γραφικών **NVIDIA RTX A5000**, καθεμία εξοπλισμένη με **24GB VRAM**. Η χρήση αυτών των

[1]Το WordNet είναι μια μεγάλης κλίμακας, ιεραρχική βάση δεδομένων λεξιλογικών όρων, όπου οι λέξεις οργανώνονται σε ομάδες συνώνυμων (synsets)

καρτών γραφικών κρίθηκε απαραίτητη ώστε να αντιμετωπιστούν οι υψηλές υπολογιστικές απαιτήσεις των μοντέλων και του μεγάλου συνόλου δεδομένων που χρησιμοποιήθηκαν. Παρακάτω, παρουσιάζονται επιγραμματικά ορισμένα σημαντικά χαρακτηριστικά τους.

Διαμόρφωση Hardware

- **Πυρήνες:** 8192 CUDA πυρήνες παράλληλης επεξεργασίας
- **Εύρος Ζώνης Μνήμης:** 768 GB/s
- **Ακρίβεια:** Υποστηρίζει λειτουργίες FP32, FP16 και INT8

Η δομή της προτεινόμενης μεθοδολογίας εξετάζεται σε τρία (3) διαφορετικά μοντέλα (frameworks), το MAE, το DINO και το DINOv2. Πριν την εφαρμογή της προσεκτικής χρήσης (attentive probing), ήταν απαραίτητο να επαναπροσδιοριστούν εκ νέου τα αποτελέσματα της γραμμικής χρήσης χαρακτηριστικών (linear probing), τα οποία αντιστοιχούν στα προ-εκπαιδευμένα μοντέλα όπως ακριβώς αυτά ορίζονται στα επίσημα αποθετήρια του κάθε πλαισίου. Σκοπός είναι να επιβεβαιωθεί η σωστή λειτουργία και χρήση των μοντέλων καθώς και η διαφάνεια των αποτελεσμάτων στη προαναφερθείσα δομή καρτών γραφικών.

Τόσο ο επαναπροσδιορισμός των αποτελεσμάτων της γραμμικής χρήσης (linear probing) όσο και τα πειράματα της προτεινόμενης μεθοδολογίας (attentive probing) που θα περιγραφούν εκτενώς παρακάτω, πραγματοποιήθηκαν με τα προ-εκπαιδευμένα μοντέλα των εξής αποθετηρίων:

- Για το MAE χρησιμοποιήθηκε προ-εκπαιδευμένο δίκτυο αρχιτεκτονικής ViT-Base από το επίσημο αποθετηρίο της *facebook research*: (GitHub Repository)
- Για το DINO χρησιμοποιήθηκε προ-εκπαιδευμένο δίκτυο αρχιτεκτονικής ViT-Small από το επίσημο αποθετηρίο του «Keep It SimPool: Who Said Supervised Transformers Suffer from Attention Deficit?» [20] (GitHub Repository)
- Για το DINOv2 χρησιμοποιήθηκε προ-εκπαιδευμένο δίκτυο αρχιτεκτονικής ViT-Small από το επίσημο αποθετηρίο της *facebook research*: (GitHub Repository)

Πειραματική Διάταξη Μοντέλου MAE

Όλα τα πειράματα πραγματοποιήθηκαν για 90 εποχές εκπαίδευσης, ενώ το μέγεθος κάθε παρτίδας (batch size) ορίζεται ως εξής:

$$EBS = 512(batch\ size\ per\ gpu) \times 16(accum\ iter) \times 2(GPUs) \tag{5.1}$$

όπου *EBS* είναι το ισχύον μέγεθος παρτίδας (effective batch size), *512* είναι ο αριθμός παρτίδας σε κάθε GPU, *16* οι συσσωρευμένες επαναλήψεις και *2* ο αριθμός των GPUs που χρησιμοποιούνται παράλληλα. Ως αποτέλεσμα, το συνολικό μέγεθος ανά παρτίδα κατά τη διάρκεια της εκπαίδευσης ανέρχεται στα ***16384*** δείγματα.

Ο ρυθμός εκμάθησης υπολογίζεται με βάση τον κανόνα γραμμικής κλιμάκωσης και δίνεται από την εξίσωση:

$$lr = 1e^{-1} \times \frac{EBS}{256} \tag{5.2}$$

όπου *EBS* είναι το ισχύον μέγεθος παρτίδας (effective batch size), $1e^{-1}$ είναι ο βασικός ρυθμός εκμάθησης (Base Learning Rate - blr) και χρησιμοποιείται το μέγεθος που έχει προκαθοριστεί από τους δημιουργούς. Το μοντέλο οπτικού μετασχηματιστή που χρησιμοποιείται για την αναπαραγωγή του μοντέλου είναι το ViT-Base 16 τμημάτων (patches) και ο αλγόριθμος βελτιστοποίησης είναι ο LARS. Το μοντέλο κατά κανόνα χρησιμοποιεί είτε τα χαρακτηριστικά ή τη μονάδα ταξινόμησης «cls token», αποκλειστικά από το τελευταίο τμήμα-μπλοκ της εκάστοτε επιλεγμένης αρχιτεκτονικής.

Πειραματική Διάταξη Μοντέλου DINO

Σε αυτή τη περίπτωση το προ-εκπαιδευμένο δίκτυο που χρησιμοποιήθηκε, προέρχεται από το επίσημο αποθετήριο του Simpool και όχι της meta AI, με τη μόνη διαφορά να έγκειται στις εποχές εκπαίδευσης των δύο μοντέλων με 100 έναντι 800 αντίστοιχα. Η επιλογή αυτή έγινε καθώς στη σελίδα του Simpool παρέχεται ένα ακόμα προεκπαιδευμένο δίκτυο με μια μικρή παραλλαγή στην γνήσια αρχιτεκτονική, όπως θα δούμε παρακάτω, το οποίο χρησιμοποιείται κατά την ανάλυση των πειραμάτων. Για να είναι δυνατή η σύγκριση των αποτελεσμάτων μεταξύ όλων των μεθόδων στο ίδιο μοντέλο, τα προ-εκπαιδευμένα μοντέλα είναι απαραίτητο να έχουν εκπαιδευτεί υπό τις ίδιες τεχνικές προδιαγρφές.

Οι εποχές εκπαίδευσης για γραμμική και προσεκτική χρήση τέθηκαν στις *100* και το μέγεθος παρτίδας κάθε GPU ορίστηκε σε *512*, ακολουθώντας τις επίσημες προεπιλεγμένες υπερπαραμέτρους. Ο ρυθμός εκμάθησης που χρησιμοποιήθηκε είναι $1e^{-3}$, το μοντέλο οπτικού μετασχηματιστή που χρησιμοποιείται είναι το ViT-Small 16 τμημάτων (patches) και ο αλγόριθμος βελτιστοποίησης είναι με στοχαστική κάθοδο κλίσης (SGD). Το συγκεκριμένο μοντέλο επιτρέπει την αξιοποίηση τόσο του τελευταίου τμήματος της

αρχιτεκτονικής αλλά και των τελευταίων τεσσάρων (4) τμημάτων ταυτόχρονα για μεγαλύτερη και πληρέστερη αναπαράσταση με απαραίτητη (από προεπιλογή) τη χρήση της μονάδας ταξινόμησης (cls token). Περισσότερες λεπτομέρειες για την δομή κάθε πειράματα θα σημειωθούν στην επόμενη ενότητα.

Πειραματική Διάταξη Μοντέλου DINOv2

Στο πλαίσιο του DINOv2 χρησιμοποιείται το προ-εκπαιδευμένο μοντέλο, όπως δίνεται από την επίσημη ιστοσελίδα της meta AI. Όντας μια βελτιστοποιημένη μορφή του προηγούμενου πλαισίου, πειράματα γραμμικής και προσεκτικής χρήσης χαρακτηριστικών πραγματοποιούνται για πολλαπλούς ταξινομητές παράλληλα, για μόλις *10* εποχές, με το μέγεθος παρτίδας να είναι 512 ανά GPU. Στα πλαίσια ενός πειράματος οι ταξινομητές που αξιολογούνται μπορεί να ποικίλλουν με διαφορετικούς ρυθμούς εκμάθησης, στο διάστημα $[1e^{-5}, 0.1]$, με συνεχώς αυξανόμενο βήμα, στα τμήματα της αρχιτεκτονικής από τα οποία λαμβάνουμε προ-εκπαιδευμένα χαρακτηριστικά αλλά και στην χρήση ή μη του ολικού μέσου όρου (Global Average Pooling - GAP), καθώς η χρήση της μονάδας ταξινόμησης είναι κατά προεπιλογή δεδομένη. Η αρχιτεκτονική που χρησιμοποιείται είναι ViT-Small 14 τμημάτων (patches) και ο αλγόριθμος βελτιστοποίησης είναι με στοχαστική κάθοδο κλίσης (SGD).

5.3 Πειραματικά Αποτελέσματα

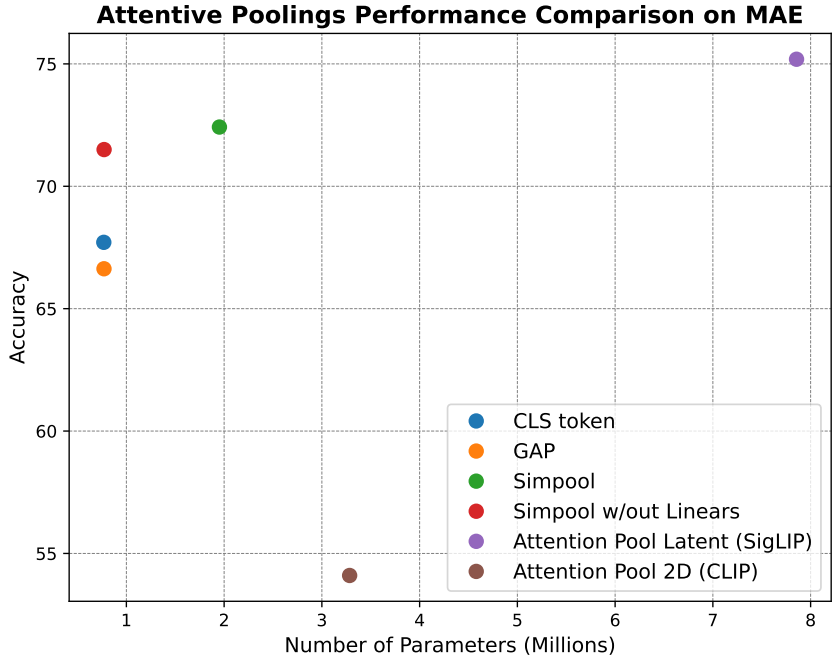
Σε αυτό το σημείο της εργασίας, παρουσιάζονται όλα τα πειραματικά αποτελέσματα, τα οποία χωρίζονται σε ποσοτικά και ποιοτικά. Στα ποσοτικά αποτελέσματα, θα αναλυθούν τα τεχνικά χαρακτηριστικά, το υπολογιστικό κόστος και οι ακρίβειες που επιτεύχθηκαν για κάθε πειραματική μέθοδο γραμμικής χρήσης και προσεκτικής χρήσης, βάσει των προ-εκπαιδευμένων μοντέλων MAE, DINO και DINOv2, αλλά θα πραγματοποιηθούν και συγκρίσεις μεταξύ αυτών. Τα ποιοτικά αποτελέσματα περιλαμβάνουν τους εξαγομένους χάρτες με τα βάρη προσοχής και θα σχολιαστούν διεξοδικά παρακάτω.

5.3.1 Ποσοτικά Αποτελέσματα

Στον πίνακα 5.1 παρουσιάζεται η σύγκριση των μεθόδων γραμμικής χρήσης (linear probing) και προσεκτικής χρήσης (attentive probing), στο σύνολο δεδομένων ImageNet 1K χρησιμοποιώντας την αρχιτεκτονική ViT-B/16 για το MAE. Σε όλα τα πειράματα του συγκεκριμένου πίνακα, αξιοποιούνται τα εξαγόμενα χαρακτηριστικά του τελευταί-

Method	Arch.	Accuracy	Params (M)
Linear Probing (GAP)	ViT-B/16	66.6%	0.769
Linear Probing (CLS)	ViT-B/16	67.7%	0.771
CLIP	ViT-B/16	54.1%	3.284
SigLIP	ViT-B/16	75.2%	7.856
Simpool w/out linears	ViT-B/16	**71.5%**	**0.772**
Simpool	ViT-B/16	**72.4%**	**1.952**

Πίνακας 5.1: Σύγκριση απόδοσης τεχνικών γραμμικής και προσεκτικής χρήσης στο ImageNet 1K για το MAE, χρησιμοποιώντας την αρχιτεκτονική ViT-B/16. Arch: architecture, Params: parameters



Σχήμα 5.1: Γραφική αναπαράσταση σύγκρισης απόδοσης διαφορετικών υφιστάμενων τεχνικών προσεκτικής χρήσης στο ImageNet 1K για το MAE, σε συνδυασμό με τις παραμέτρους προς εκπαίδευση που αντιστοιχούν στην αρχιτεκτονική ViT-B/16

ου μπλοκ του προ-εκπαιδευμένου μοντέλου, εκτός από το δεύτερο πείραμα (γραμμική χρήση με μονάδα ταξινόμησης) στο οποίο αξιοποιείται αποκλειστικά η μονάδα ταξινόμησης του τελευταίου μπλοκ. Η μέθοδος της γραμμικής χρήσης με τη μονάδα ταξινόμησης υπερτερεί αυτής του GAP κατά 1.1% ενώ και οι δύο μέθοδοι διατηρούν παρόμοιο σταθερό αριθμό παραμέτρων κοντά στο 0.77 εκατομμύρια. Οι μέθοδοι προσεκτικής χρήσης παρουσιάζουν σημαντικό ενδιαφέρον, με την καλύτερη απόδοση να επιτυγχάνεται με τη μέθοδο SigLIP φτάνοντας σε ακρίβεια 75.2% και ξεπερνώντας τις μεθόδους

γραμμικής χρήσης. Ωστόσο, δεν παρατηρείται ισορροπημένη σχέση ακρίβειας και πολυπλοκότητας καθώς ο αριθμός των παραμέτρων ξεπερνά τα 7.8 εκατομμύρια. Η μέθοδος CLIP, παρά το ότι χρησιμοποιεί αρκετά υψηλό αριθμό παραμέτρων, επιτυγχάνει σημαντικά χαμηλότερη ακρίβεια φτάνοντας στο 54.1%. Αντιθέτως, η μέθοδος Simpool επιτυγχάνει αξιοσημείωτες ακρίβειες σε συνδυασμό με πολύ χαμηλή πολυπλοκότητα παραμέτρων. Συγκεκριμένα, το Simpool χωρίς γραμμικά επίπεδα φτάνει σε ακρίβεια 71.5% με μόλις 0.772 εκατομμύρια παραμέτρους, ενώ το simpool επιτυγχάνει καλύτερη ακρίβεια 72.4% με χαμηλά επίπεδα παραμέτρων σε σύγκριση με τις μεθόδους CLIP και SigLIP, μόλις 1.952 εκατομμύρια. Τα παραπάνω αποτελέσματα υπογραμμίζουν την ισορροπημένη απόδοση σε ικανοποιητική ακρίβεια και πολυπλοκότητα της μεθόδου simpool σε σύγκριση με παρόμοιες τεχνικές.

Το σχήμα 5.1 απεικονίζει την σχέση ακρίβειας και αριθμού παραμέτρων των μεθόδων που περιγράφηκαν. Όσο πιο αριστερά στον άξονα των x και πιο πάνω στον άξονα των y βρίσκεται μια μέθοδος, τόσο καλύτερη αντισταθμιστική σχέση παρουσιάζει μεταξύ υπολογιστικού κόστους και ακρίβειας. Εδώ είναι φανερό ότι το simpool και το simpool χωρίς γραμμικά επίπεδα επιτυγχάνουν μια τέτοια σχέση, ενώ για παράδειγμα το SigLIP όπως προαναφέρθηκε, παρά το γεγονός ότι παρουσιάζει βελτιωμένη ακρίβεια, η απαίτηση σε υπολογιστικό κόστος είναι πολύ μεγάλη για αυτό συναντάται στο πάνω δεξιά τμήμα της γραφικής αναπαράστασης.

		Vectors concat. before linear evaluation			
Method	**Arch.**	**CLS**	**Att. Probing**	**n blocks**	**Accuracy**
Linear Probing	ViT-S/16	✓		1	71.5%
Simpool	ViT-S/16	✓	✓	1	71.3%
Simpool	ViT-S/16 w/ Simpool		✓	1	72.4%
Linear Probing	ViT-S/16	✓		4	72.9%
Simpool	ViT-S/16	✓	✓	4	**73.3%**

Πίνακας 5.2: Σύγκριση απόδοσης συνδυασμών τεχνικών γραμμικής και προσεκτικής χρήσης στο ImageNet 1K για το DINO, χρησιμοποιώντας την αρχιτεκτονική ViT-S/16. Arch: architecture, Att.Probing: attentive probing, n blocks: number of blocks

Στον πίνακα 5.2 παρουσιάζονται τα αποτελέσματα της σύγκρισης των γραμμικών μεθόδων χρήσης και προσεκτικής χρήσης βάση του προ-εκπαιδευμένου μοντέλου DINO. Στο πρώτο πείραμα γραμμικής χρήσης αξιοποιείται η μονάδα ταξινόμησης του τελευταίου μπλοκ του προ-εκπαιδευμένου μοντέλου και επιτυγχάνεται ακρίβεια 71.5%. Στο πεί-

ραμα που ακολουθεί, ενσωματώνεται η τεχνική προσεκτικής χρήσης στο μοντέλο, αξιοποιώντας επίσης τη μονάδα ταξινόμησης του τελευταίου μπλοκ, φτάνοντας σε χαμηλότερη ακρίβεια μεγέθους 71.3%. Με την αρχιτεκτονική Vit-S/16 που έχει προ-εκπαιδευτεί με την χρήση της μεθόδου Simpool και αξιοποιεί όλα τα εξαγόμενα χαρακτηριστικά του τελευταίου του μπλοκ με Simpool τεχνική, επιτυγχάνεται 72.4% ακρίβεια. Τέλος, πραγματοποιήθηκαν δύο καθοριστικά πειράματα για την εξαγωγή συμπερασμάτων, τα οποία είναι πανομοιότυπα με τα πρώτα δύο πειράματα του πίνακα, με την βασική διαφορά ότι αξιοποιείται η μονάδα ταξινόμησης και τα εξαγόμενα χαρακτηριστικά του προ-εκπαιδευμένου μοντέλου από τα τέσσερα τελευταία του μπλοκ. Σε αυτά, συμπεραίνεται ότι το πείραμα με Simpool, επιτυγχάνει την μεγαλύτερη ακρίβεια, μεγέθους 73.3%.

		Vectors concat. before linear evaluation			
Method	**Arch.**	**CLS (n blocks)**	**GAP (n blocks)**	**Att. Probing**	**Accuracy**
Linear Probing	ViT-S/14	✓			80.5%
Linear Probing	ViT-S/14	✓(4)			80.9%
Linear Probing	ViT-S/14		✓		77.9%
Linear Probing	ViT-S/14		✓(4)		78.6%
Simpool w/out linears	ViT-S/14			✓	80.0%
Simpool	ViT-S/14			✓	81.1%
Simpool	ViT-S/14			✓(4)	81.6%
4x Simpool	ViT-S/14			✓(4)	81.9%
Linear Probing	ViT-S/14	✓	✓		81.0%
Linear Probing	ViT-S/14	✓(4)	✓		81.1%
Simpool	ViT-S/14	✓		✓	81.5%
Simpool	ViT-S/14	✓(4)		✓	81.7%
Simpool	ViT-S/14	✓	✓	✓	81.7%
Simpool	ViT-S/14	✓(4)	✓	✓	81.8%

Πίνακας 5.3: Σύγκριση απόδοσης τεχνικών γραμμικής και προσεκτικής χρήσης στο ImageNet 1K για το DINOv2, χρησιμοποιώντας την αρχιτεκτονική ViT-S/14, με διαφορετικούς τρόπους αξιοποίησης των ενός (1) ή τεσσάρων (4) τελευταίων τμημάτων του προ-εκπαιδευμένου μοντέλου. Arch: architecture, (n blocks): number of blocks, Att.Probing: attentive probing

Ο πίνακας 5.3 παρουσιάζει πολλαπλά πειράματα γραμμικής χρήσης και προσεκτικής χρήσης βάσει του προ-εκπαιδευμένου μοντέλου DINOv2 με αρχιτεκτονική ViT-S/14, το οποίο αποτελεί μια βελτιστοποιημένη μορφή του μοντέλου DINO. Η αρχιτεκτονική Vit-Small επιλέχθηκε λόγω περιορισμών των υπολογιστικών πόρων, καθώς δεν είναι

υπερβολικά βαθιά και επιτρέπει την επεξεργασία εικόνας με λιγότερη υπολογιστική ισχύ και μνήμη, διατηρώντας παράλληλα υψηλή απόδοση και ακρίβεια.

Τα αποτελέσματα που καταγράφηκαν είναι βάση των βέλτιστων ρυθμών εκμάθησης. Σε πρώτη φάση, πραγματοποιούνται δύο πειράματα γραμμικής χρήσης, το πρώτο αξιοποιώντας αποκλειστικά τη μονάδα ταξινόμησης τελευταίου μπλοκ και το δεύτερο, των τεσσάρων τελευταίων μπλοκ του προ-εκπαιδευμένου μοντέλου. Καλύτερη απόδοση για 0.4% παρουσίασε το δεύτερο πείραμα, επιτυγχάνοντας ακρίβεια 80.9% έναντι του πρώτου με 80.5%. Στη συνέχεια, δύο ακόμα πειράματα γραμμικής χρήσης υλοποιούνται, το πρώτο αξιοποιώντας τον μέσο όρο των εξαγόμενων χαρακτηριστικών του τελευταίου μπλοκ και το δεύτερο τον μέσο όρο των εξαγόμενων χαρακτηριστικών των τεσσάρων τελευταίων μπλοκ του προ-εκπαιδευμένου μοντέλου (μέθοδος GAP), αγνοώντας το διάνυσμα της μονάδας ταξινόμησης. Όπως και στα προηγούμενα δύο πειράματα, η αξιοποίηση της πληροφορίας στα τέσσερα τελευταία μπλοκ οδηγεί σε βελτιωμένη ακρίβεια 78.6% έναντι της περίπτωσης που χρησιμοποιείται αποκλειστικά το τελευταίο μπλοκ, επιτυγχάνοντας ακρίβεια 77.9%. Τέσσερα διαδοχικά πειράματα με Simpool, διαφορετικής διάταξης, πραγματοποιούνται στην συνέχεια. Η μέθοδος simpool χωρίς γραμμικά επίπεδα, η οποία αξιοποιεί τα εξαγόμενα χαρακτηριστικά του τελευταίου μπλοκ, επιτυγχάνει ακρίβεια της τάξεως του 80.0%.

Ακολουθούν δύο πειράματα της μεθόδου από τα οποία το ένα αξιοποιεί τα εξαγόμενα χαρακτηριστικά από το τελευταίο μπλοκ ενώ το δεύτερο, από τα τέσσερα τελευταία. Καλύτερη απόδοση μεταξύ των δύο αυτών, όπως ήταν αναμενόμενο, φαίνεται να δείχνει η διάταξη με τα τέσσερα μπλοκ με 81.6% έναντι από αυτής με το ένα με ακρίβεια 81.1%. Στο τέταρτο πείραμα έχει υλοποιηθεί η μέθοδος Simpool τέσσερις φορές στα τέσσερα τελευταία μπλοκ, με την ταυτόχρονη υλοποίηση τεσσάρων δικτύων Simpool. Σε αυτή την περίπτωση το υπολογιστικό κόστος τετραπλασιάζεται καθώς για κάθε μπλοκ αναλογούν διαφορετικοί πίνακες βαρών. Το συγκεκριμένο επιτυγχάνει καλύτερη ακρίβεια από όλα τα προηγούμενα, όμως με τετραπλάσιο αριθμό παραμέτρων. τα δύο τελευταία πειράματα παρουσιάζουν παρόμοια αρχιτεκτονική με την διαφορά τους να έγκειται στον διαμοιρασμό ή μη των προς εκπαίδευση παραμέτρων μεταξύ των τεσσάρων τελευταίων μπλοκ. Αξίζει να σημειωθεί πως η διαφορά τους είναι μόλις 0.3% επιδεικνύοντας τη σταθερότητα της αρχιτεκτονικής και δίχως τον τετραπλασιασμό των παραμέτρων.

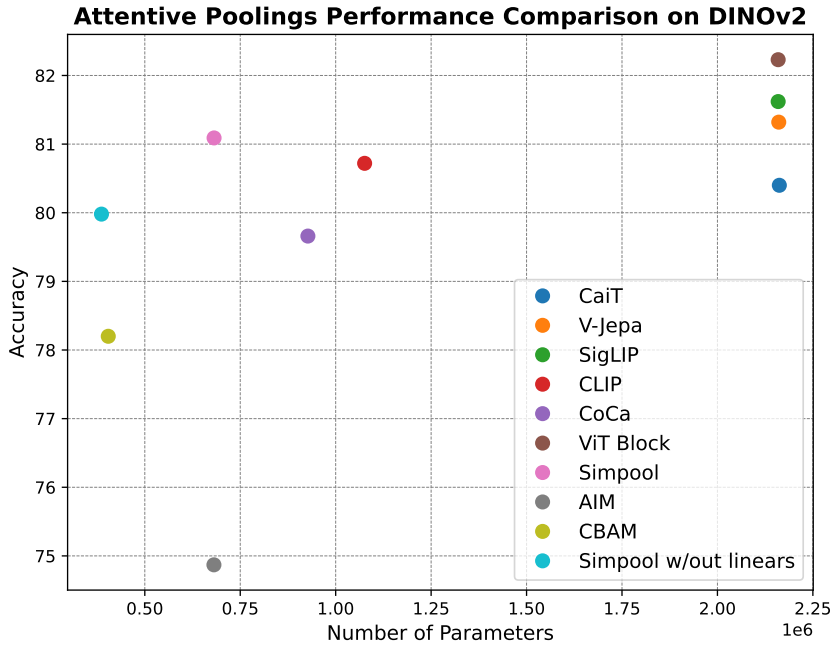
Στην συνέχεια, ακολουθεί μια σειρά πειραματικών διαδικασιών αρκετά διαφορετικής διάταξης με όλα τα προηγούμενα. Ένα πείραμα γραμμικής χρήσης, αξιοποιώντας τη μονάδα ταξινόμησης και το ολικό μέσο όρο (GAP) του τελευταίου μπλοκ του προ-εκπαιδευμένου μοντέλου, επιτυγχάνει ακρίβεια 81.0%. Το επόμενο, διαφέρει μόνο στο

ότι αξιοποιεί τη μονάδα ταξινόμησης των τεσσάρων τελευταίων μπλοκ και επιτυγχάνει 0.1% παραπάνω ακρίβεια. Δύο πειράματα με μέθοδο Simpool, το ένα αξιοποιώντας τη μονάδα ταξινόμησης του τελευταίου μπλοκ και το άλλο των τεσσάρων τελευταίων, επιτυγχάνουν 81.5% και 81.7% αντίστοιχα και ξεπερνούν τα δύο προαναφερθέντα, άμεσα συγκρίσιμα πειράματα. Τέλος, υλοποιούνται δύο ακόμα με την μέθοδο Simpool με την εξής διάταξη: Το πρώτο αξιοποιεί μονάδα ταξινόμησης και ολικό μπλοκ σε συνδυασμό με τον ολικό μέσο όρο. Στις δύο αυτές υλοποιήσεις, το αποτέλεσμα απόδοσης τους διαφέρει μόνο για 0.1%, φτάνοντας σε ακρίβειες 81.7% και 81.8% αντίστοιχα. Για ακομα μια φορά επιδεικνύεται η ανωτερότητα της προτεινόμενης αρχιτεκτονικής έναντι του τυπικού γενικευμένου μέσου όρου βελτιώνοντας την ακρίβεια σε κάθε περίπτωση.

Pooling Method	ViT-Small Params	Accuracy	Scaled LR
CaiT	2,162,152	80.40%	0.08
V-Jepa	2,160,616	81.32%	0.2
ViT Block	2,159,080	**82.23%**	0.4
SigLIP	2,159,080	81.62%	0.2
CLIP	1,075,816	80.72%	0.08
CoCa	927,208	79.66%	0.08
Simpool	681,448	81.09%	0.2
AIM	681,064	74.87%	0.4
CBAM	404,300	78.20%	0.2
Simpool w/out linears	**386,536**	79.98%	0.4

Πίνακας 5.4: Σύγκριση απόδοσης διαφορετικών υφιστάμενων τεχνικών προσεκτικής χρήσης στο ImageNet 1K για το DINOv2, σε συνδυασμό με τις παραμέτρους προς εκπαίδευση που αντιστοιχούν στην αρχιτεκτονική ViT-S/14. Scaled LR: scaled learning rate

Στον πίνακα 5.4 και στο σχήμα 5.2 παρουσιάζονται αριθμητικά και γραφικά, οι συγκρίσεις απόδοσης και απαιτούμενου υπολογιστικού κόστους βάση της αρχιτεκτονικής ViT-S/14, διαφορετικών υφιστάμενων τεχνικών με προσεκτική χρήση, βάσει του DINOv2. Η μέθοδος SigLIP και ViT Block, επιτυγχάνουν τις καλύτερες ακρίβειες σε σχέση με τις υπόλοιπες μεθόδους με την δεύτερη να έχει την υψηλότερη επιτυγχάνοντας 82.23%, όμως με πολύ επιβαρυμένο υπολογιστικό κόστος το οποίο φτάνει στις 2,159,080 παραμέτρους και για τις δύο. Αξιοσημείωτο είναι το γεγονός ότι το Simpool και το simpool χωρίς γραμμικά επίπεδα επιτυγχάνουν σημαντικές ακρίβειες σε συνδυασμό με αρκετά χαμηλό υπολογιστικό κόστος με το δεύτερο να έχει τον πιο μικρό αριθμό σε παραμέτρους, σε σύγκριση με όλες τις υπόλοιπες μεθόδους. Μπορεί να παρατηρηθεί ότι η μέθοδος Vit Block, η οποία αντιστοιχεί σε έναν κωδικοποιητή του οπτικού μετασχημα-



Σχήμα 5.2: Γραφική αναπαράσταση σύγκρισης απόδοσης διαφορετικών υφιστάμενων τεχνικών προσεκτικής χρήσης στο ImageNet 1K για το DINOv2, σε συνδυασμό με τις παραμέτρους προς εκπαίδευση που αντιστοιχούν στην αρχιτεκτονική ViT-S/14

τιστή , επιτυγχάνει την ίδια ακρίβεια με το Simpool αλλά με 148,224 παραπάνω παραμέτρους και οι μέθοδοι CLIP και SigLIP που έχουν παρόμοια αρχιτεκτονική με τον Simpool απαιτούν αρκετά περισσότερες παραμέτρους. Μέθοδοι όπως το CaiT και το V-Jepa φτάνουν σε αρκετά ικανοποιητικές ακρίβειες, όμως αποδεικνύονται από τις πιο 'ακριβές' υπολογιστικά μεθόδους με πάνω από 2,160,000 παραμέτρους. Η μέθοδος CoCa απαιτεί μεγαλύτερο αριθμό παραμέτρων από το Simpool και αποδίδει χειρότερα από αυτό με ακρίβεια που φτάνει στο 79.66% έναντι του Simpool και simpool χωρίς γραμμικά επίπεδα με ακρίβειες 81.09% και 79.98% αντίστοιχα. Χειρότερη απόδοση σε σύγκριση με όλες τις υπόλοιπες παρουσίασε η μέθοδος AIM με την ακρίβεια του να φτάνει το 74.87% με ελάχιστα πιο λίγες απαιτούμενες παραμέτρους από το Simpool. Τέλος, η μέθοδος CBAM βρίσκεται χαμηλά στην κατάταξη του απαιτούμενου υπολογιστικού κόστους με μόλις 404,300 παραμέτρους επιτυγχάνοντας χαμηλότερη επίσης ακρίβεια και από τις δύο μεθόδους Simpool, η οποία φτάνει το 78.20%.

5.3.2 Ποιοτικά Αποτελέσματα

Στα σχήματα 5.3 και 5.4 απεικονίζονται οι χάρτες των βαρών προσοχής βάσει του μοντέλου **MAE**. Σε αυτούς τους χάρτες απεικονίζονται τα βάρη που έχει δώσει το μοντέλο

για τον εντοπισμό των πιο σημαντικών χαρακτηριστικών, υπογραμμίζοντας με κίτρινο τα εικονοστοιχεία που θεωρεί πιο σημαντικά. Οι στήλες αντιπροσωπεύουν τις διαφορετικές μεθόδους και συγκεκριμένα, η πρώτη αφορά τα εξαγόμενα χαρακτηριστικά από το τελευταίο μπλοκ του προ-εκπαιδευμένου μοντέλου του MAE, η δεύτερη αφορά τα εξαγόμενα χαρακτηριστικά από την προσεκτική χρήση που χρησιμοποιεί η μέθοδος SigLIP που πέτυχε την καλύτερη ακρίβεια (όπως αναλύθηκε και προηγουμένως), η τρίτη και τέταρτη στήλη απεικονίζουν την προτεινόμενη τεχνική Simpool με και χωρίς γραμμικά επίπεδα και τέλος, απεικονίζεται η αυθεντική εικόνα εισόδου. Κάθε γραμμή αντιστοιχεί σε μια συγκεκριμένη εικόνα εισόδου, οι οποίες επιλέχθηκαν ώστε να ποικίλουν σε διακριτικότητα και είναι ίδιες για κάθε προ-εκπαιδευμένο πλαίσιο που μελετάται παρακάτω στα επόμενα σχήματα. Ορισμένες από αυτές έχουν διακριτό αντικείμενο προς εντοπισμό σε σχέση με το καθαρό παρασκήνιο, άλλες από αυτές περιέχουν πιο περίπλοκα μοτίβα με αρκετή πληροφορία στο παρασκήνιο όπου ο εντοπισμός του αντικειμένου προς ταξινόμηση καθίστανται αρκετά πιο δύσκολος, όπως για παράδειγμα στην έκτη γραμμή του 5.3 με τον άνθρωπο που κρατάει ένα ψάρι το οποίο είναι και το αντικείμενο προς ταξινόμηση και το παρασκήνιο είναι πλούσιο σε πληροφορία. Άλλες από αυτές περιέχουν αντικείμενα κοντινής παλέτας με το παρασκήνιο και επομένως πιο δύσκολα διαχωρίσιμα όπως για παράδειγμα η εικόνα με τους τρεις καρχαρίες στην τελευταία γραμμή του 5.3.

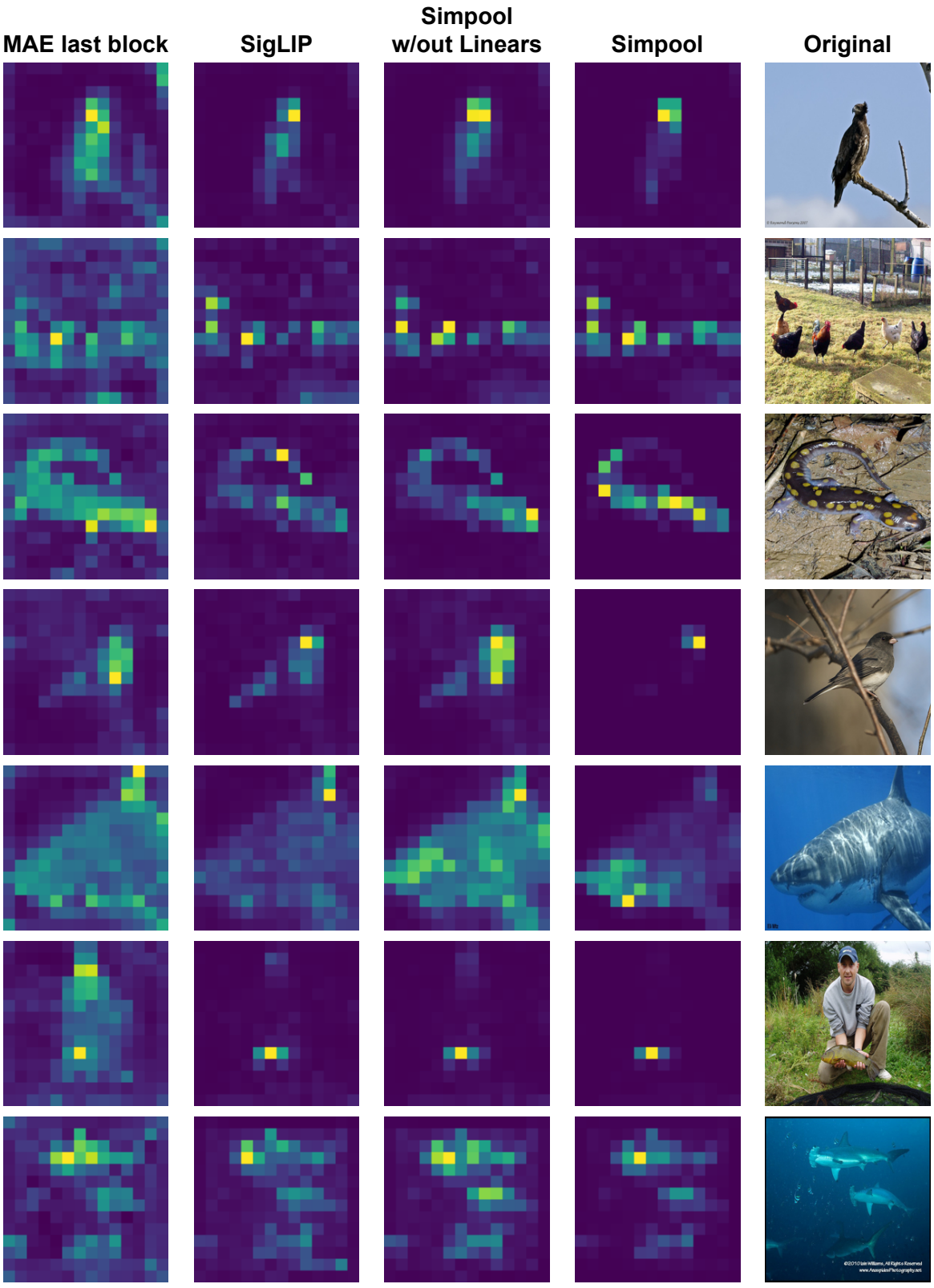
Με την παρατήρηση των δύο αυτών σχημάτων προκύπτει ότι καλύτερες αναπαραστάσεις φαίνεται να δίνει η μέθοδος Simpool. Χαρακτηριστικό παράδειγμα είναι η εικόνα με τον άνθρωπο και το ψάρι στο οποίο το Simpool έχει σωστά εντοπίσει μόνο το ψάρι ενώ στις άλλες τεχνικές, εκτός από αυτό έχουν εντοπιστεί και άλλα εικονοστοιχεία τα οποία αποτελούν θόρυβο με χαρακτηριστικό αποτέλεσμα αυτό του MAE στο τελευταίο μπλοκ. Γενικότερα, τα χειρότερα αποτελέσματα στις αναπαραστάσεις δίνει το MAE στο τελευταίο μπλοκ, εντοπίζοντας πολύ θόρυβο σε αυτές. Αξιοσημείωτα είναι τα αποτελέσματα της μεθόδου SigLIP που παρά το ότι ξεπερνά όλες τις μεθόδους σε ακρίβεια, οι χάρτες βαρών προσοχής δεν δείχνουν να εντοπίζουν μόνο το ενδιαφερόμενο αντικείμενο αλλά εντοπίζει και σημεία τα οποία δεν αντιστοιχούν στον στόχο. Χαρακτηριστικά σημεία αποτελούν τα παραδείγματα 2 και 5 του σχήματος 5.3 και 4 και 7 του σχήματος 5.4 των οποίων τα αποτελέσματα εισάγουν μεγάλο ποσοστό θορύβου.

Στο σχήμα 5.5 παρουσιάζονται οι χάρτες βαρών προσοχής για το μοντέλο **DINO**, συγκρίνοντας μόνο δύο μεθόδους. Η πρώτη είναι η εξαγωγή των σημαντικών χαρακτηριστικών από το τελευταίο block του προ-εκπαιδευμένου μοντέλου DINO και η δεύτερη είναι η μέθοδος Simpool. Τα συμπεράσματα που προκύπτουν από αυτές τις αναπα-

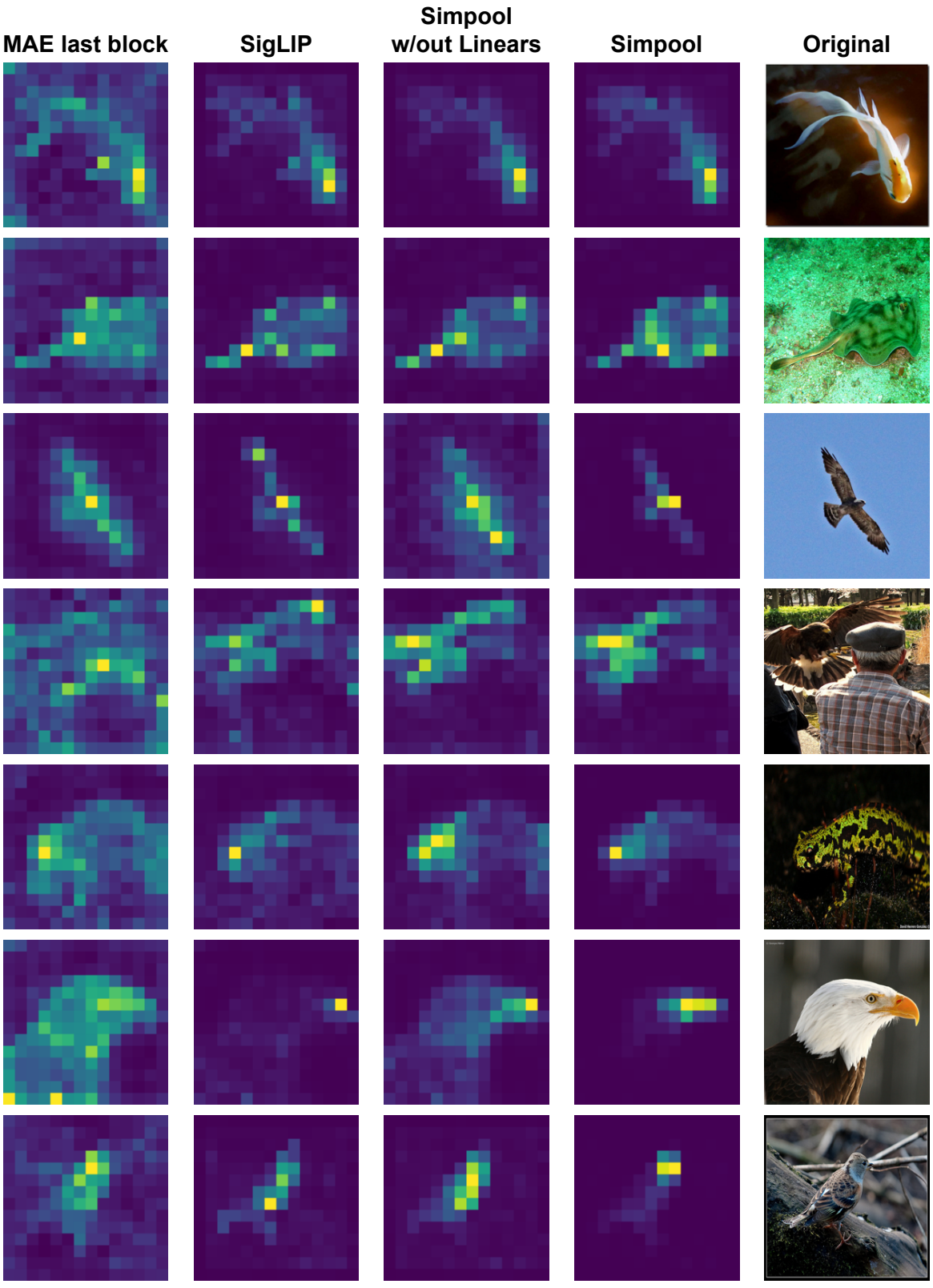
ραστάσεις είναι πως οι αποδόσεις των δύο μεθόδων είναι πολύ κοντινές. Σε ορισμένες περιπτώσεις που υπάρχει μεγαλύτερη πολυπλοκότητα, η μέθοδος Simpool εντοπίζει χαρακτηριστικά σημεία χωρίς την εισαγωγή πολύ θορύβου και είναι πιο διακριτά έναντι της άλλης μεθόδου, όπως για παράδειγμα στην δεύτερη στήλη και δεύτερη γραμμή. Ωστόσο, εντοπίζονται και περιπτώσεις που η πρώτη μέθοδος παρουσιάζει καλύτερες αναπαραστάσεις από τη μέθοδο Simpool.

Τα σχήματα 5.6 και 5.7 παρουσιάζουν τους χάρτες βαρών προσοχής για το **DINOv2** και αφορούν τέσσερις διαφορετικές μεθόδους. Οι πρώτες δύο παρουσιάζουν αναπαραστάσεις από τα εξαγόμενα χαρακτηριστικά του τελευταίου μπλοκ της προεκπαιδευμένης μεθόδου DINOv2 με ViT-S και με ViT-B αντίστοιχα. Οι άλλες δύο μέθοδοι είναι με Simpool και Simpool χωρίς γραμμικά επίπεδα. Είναι φανερό ότι και οι δύο μέθοδοι Simpool παράγουν καλύτερες αναπαραστάσεις από ότι το προ-εκπαιδευμένο μοντέλο με τις δύο αρχιτεκτονικές. Σε αυτά, δεν υπάρχει διακριτότητα στα αντικείμενα προς εντοπισμό και υπάρχει αυξημένος θόρυβος. Μεταξύ των δύο μεθόδων Simpool, αυτή με τα γραμμικά επίπεδα φαίνεται να έχει καλύτερα αποτελέσματα στις αναπαραστάσεις, εισάγοντας λιγότερο θόρυβο και καλύτερο εντοπισμό των αντικειμένων σε πολύπλοκα μοτίβα.

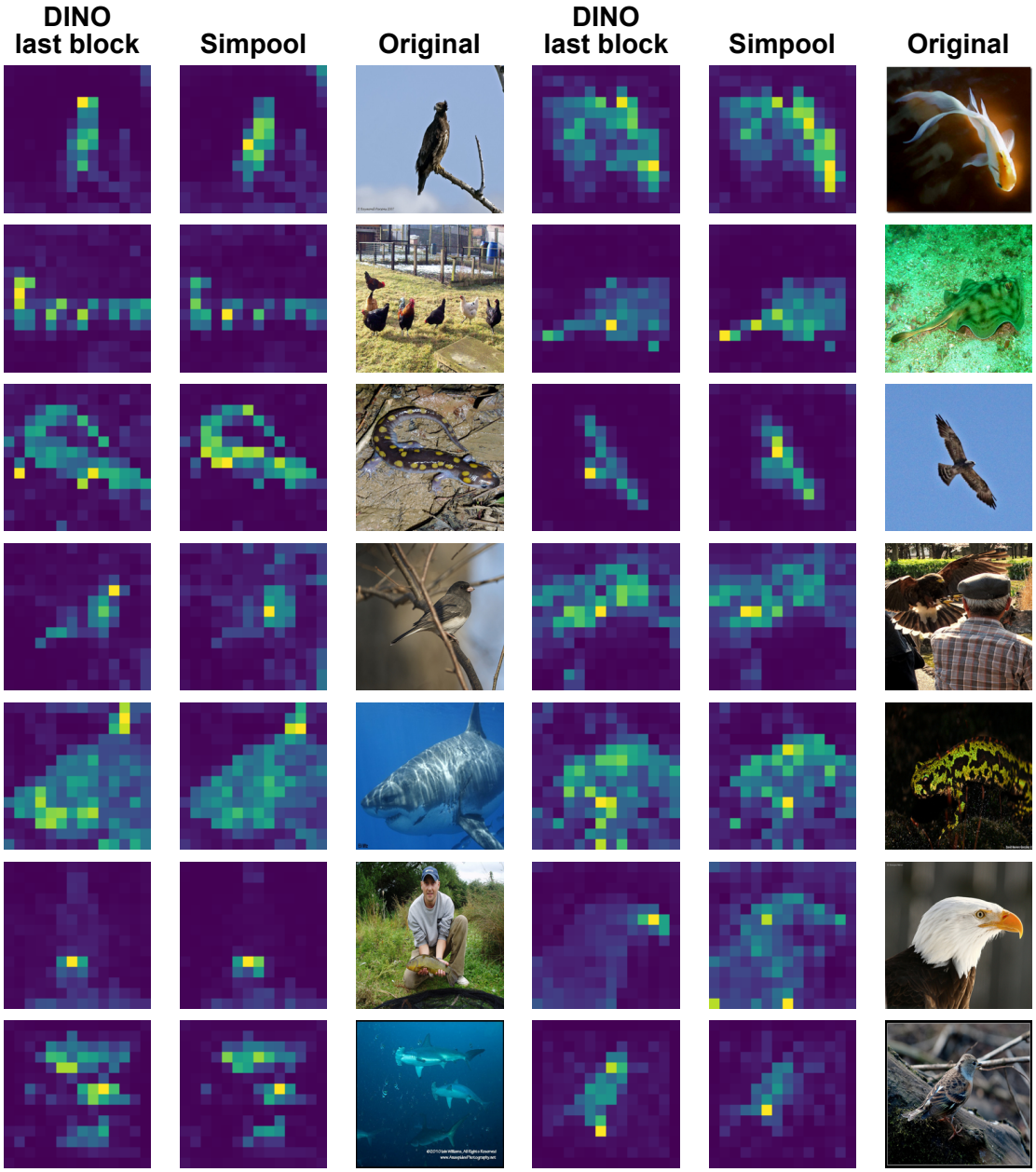
Ύστερα από την σύγκριση των αναπαραστάσεων που προκύπτουν από τα τρία προεκπαιδευμένα πλαίσια, το μοντέλο MAE με την μέθοδο **Simpool** δείχνει να αποφέρει τα καλύτερα αποτελέσματα στις αναπαραστάσεις των βαρών προσοχής. Αυτό οφείλεται στο πόσο ”καλά” χαρακτηριστικά εξάγονται από το προ-εκπαιδευμένο μοντέλο. Σε κάθε περίπτωση, η μέθοδος Simpool μπορεί να χαρακτηριστεί από ευελιξία και σταθερότητα, καθώς και στις τρεις περιπτώσεις βελτιώνει τις αναπαραστάσεις σε σημαντικό βαθμό σε σχέση με τα αποτελέσματα των αναπαραστάσεων των υπόλοιπων μεθόδων.



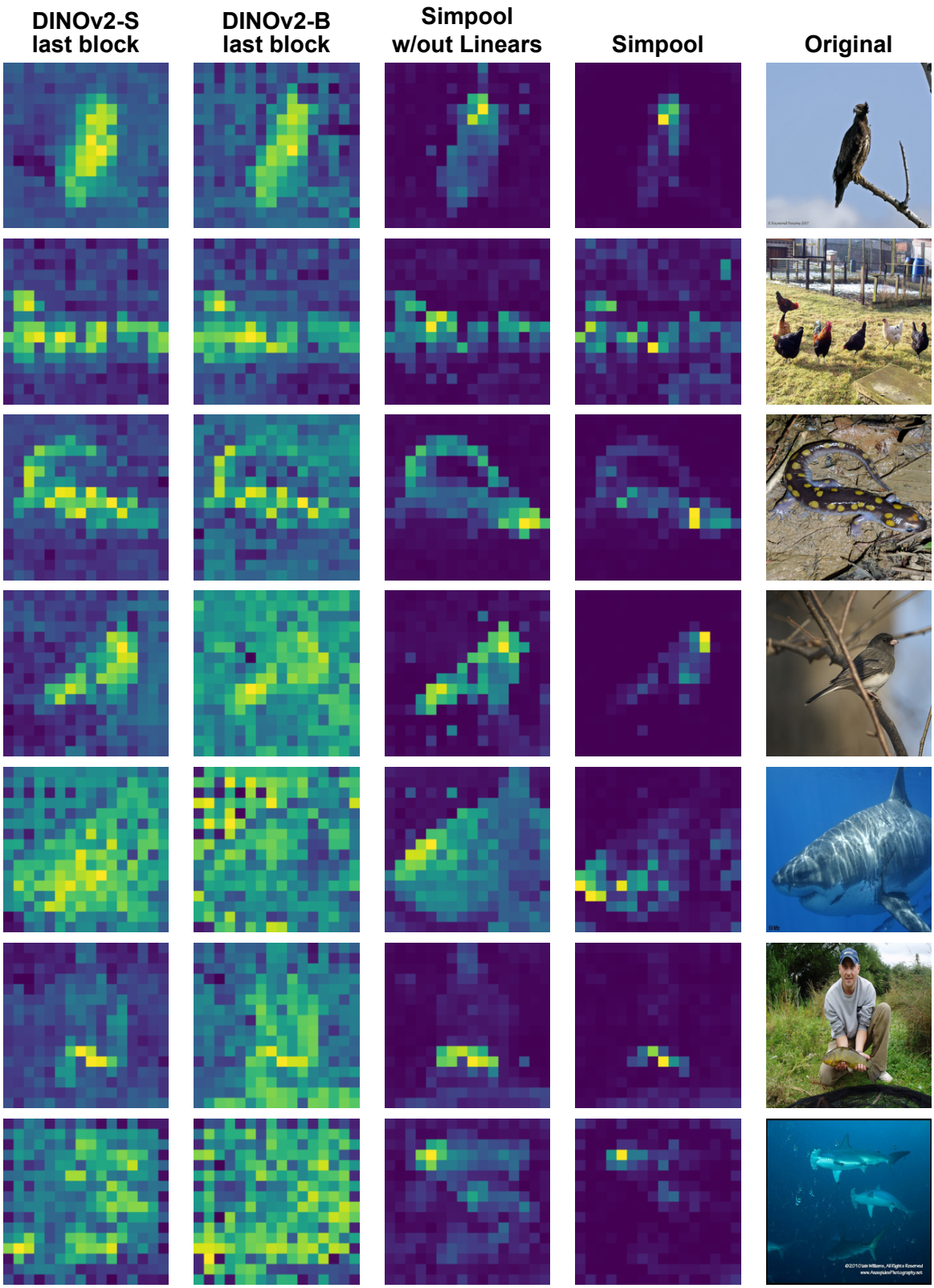
Σχήμα 5.3: Οπτικοποίηση των βαρών προσοχής (attention maps) της αρχιτεκτονικής ViT-B για το μοντέλο MAE με εκπαίδευση στο ImageNet-1k για 90 εποχές. Για το σημείο αναφοράς (baseline) ViT-B, χρησιμοποιείται ο μέσος όρος της μονάδας ταξινόμησης. Για τις υπόλοιπες μεθόδους χρησιμοποιείται η έξοδος κάθε τεχνικής συγκέντρωσης. Ανάλυση εικόνας εισόδου: 224×224, τμήματα: 16×16, χάρτες προσοχής: 14×14.



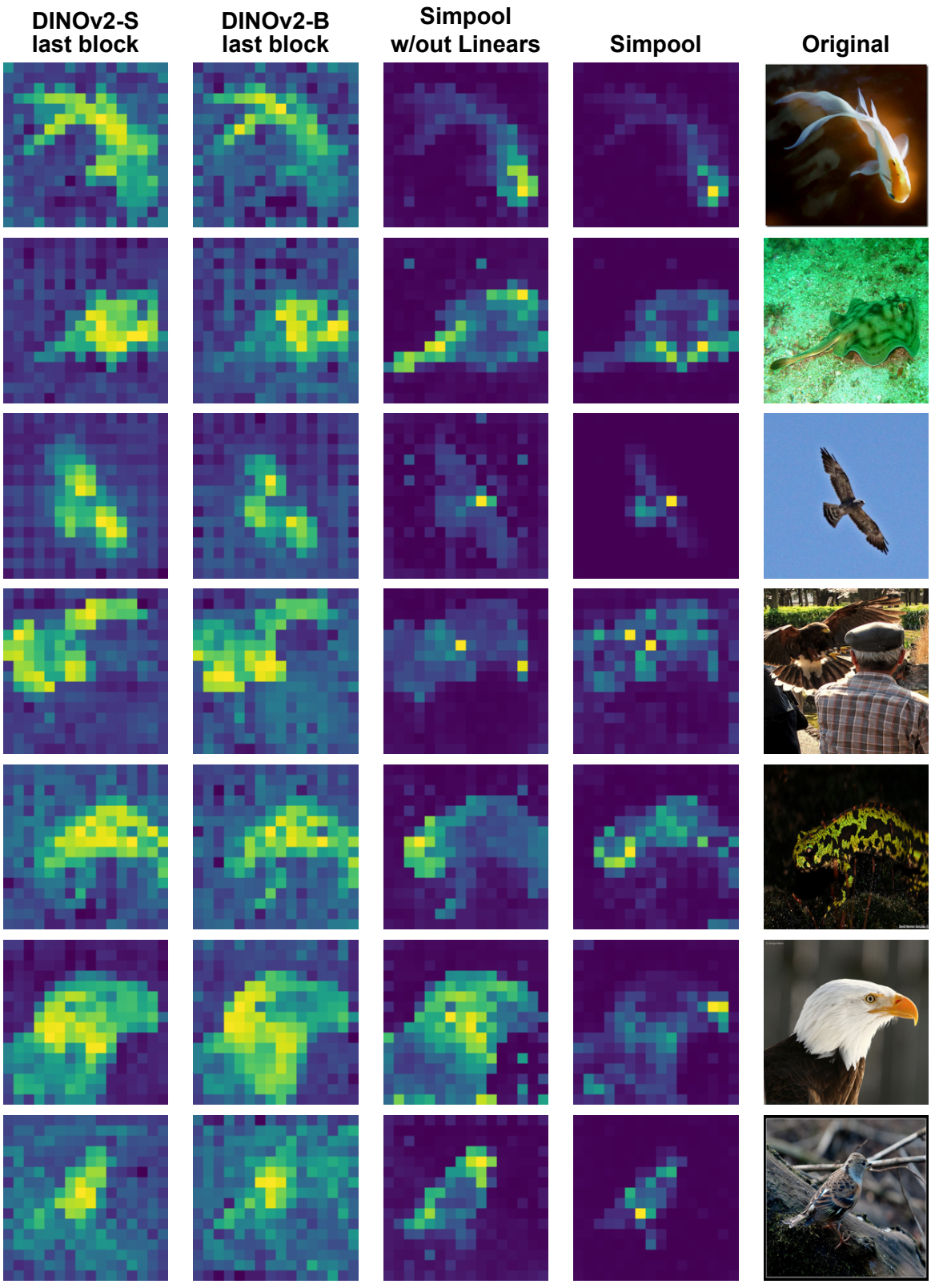
Σχήμα 5.4: Οπτικοποίηση των βαρών προσοχής (attention maps) της αρχιτεκτονικής ViT-B για το μοντέλο MAE με εκπαίδευση στο ImageNet-1k για 90 εποχές. Για το σημείο αναφοράς (baseline) ViT-B, χρησιμοποιείται ο μέσος όρος της μονάδας ταξινόμησης. Για τις υπόλοιπες μεθόδους χρησιμοποιείται η έξοδος κάθε τεχνικής συγκέντρωσης. Ανάλυση εικόνας εισόδου: 224×224, τμήματα: 16×16, χάρτες προσοχής: 14×14. (συνέχεια)



Σχήμα 5.5: Οπτικοποίηση των βαρών προσοχής (attention maps) της αρχιτεκτονικής ViT-S για το μοντέλο DINO με εκπαίδευση στο ImageNet-1k για 100 εποχές. Για το σημείο αναφοράς (baseline) ViT-S, χρησιμοποιείται ο μέσος όρος της μονάδας ταξινόμησης. Για τις υπόλοιπες μεθόδους χρησιμοποιείται η έξοδος κάθε τεχνικής συγκέντρωσης. Ανάλυση εικόνας εισόδου: 224×224, τμήματα: 16×16, χάρτες προσοχής: 14×14.



Σχήμα 5.6: Οπτικοποίηση των βαρών προσοχής (attention maps) της αρχιτεκτονικής ViT-S για το μοντέλο DINO με εκπαίδευση στο ImageNet-1k για 10 εποχές. Για το σημείο αναφοράς (baseline) ViT-S, χρησιμοποιείται ο μέσος όρος της μονάδας ταξινόμησης. Για τις υπόλοιπες μεθόδους χρησιμοποιείται η έξοδος κάθε τεχνικής συγκέντρωσης. Ανάλυση εικόνας εισόδου: 224×224, τμήματα: 14×14, χάρτες προσοχής: 16×16.



Σχήμα 5.7: Οπτικοποίηση των βαρών προσοχής (attention maps) της αρχιτεκτονικής ViT-S για το μοντέλο DINO με εκπαίδευση στο ImageNet-1k για 10 εποχές. Για το σημείο αναφοράς (baseline) ViT-S, χρησιμοποιείται ο μέσος όρος της μονάδας ταξινόμησης. Για τις υπόλοιπες μεθόδους χρησιμοποιείται η έξοδος κάθε τεχνικής συγκέντρωσης. Ανάλυση εικόνας εισόδου: 224×224, τμήματα: 14×14, χάρτες προσοχής: 16×16. (συνέχεια)

ΚΕΦΑΛΑΙΟ 6

Συμπεράσματα και Μελλοντικές Κατευθύνσεις

6.1 Συμπεράσματα

Στην παρούσα διπλωματική εργασία αναπτύξαμε μια καινοτόμο προσέγγιση στην μη γραμμική χρήση χαρακτηριστικών με μηχανισμό προσοχής, εισάγοντας την μέθοδο Simpool ως μια αποδοτική εναλλακτική της παραδοσιακής γραμμικής χρήσης χαρακτηριστικών. Μέσω των πειραμάτων που πραγματοποιήθηκαν στα προ-εκπαιδευμένα, αυτο-επιβλεπόμενα μοντέλα (MAE, DINO, DINOv2), διαπιστώθηκε ότι η προτεινόμενη μέθοδος υπερβαίνει σε απόδοση τις παραδοσιακές τεχνικές γραμμικής χρήσης καθώς και άλλες τεχνικές γραμμικής χρήσης με μηχανισμό προσοχής. Αυτό οφείλεται στην απλή αρχιτεκτονική που έχει αναπτυχθεί, χρησιμοποιώντας διασταυρούμενη προσοχή και απλουστευμένες τεχνικές που εισάγουν μικρό αριθμό παραμέτρων. Επομένως, ένα από τα βασικά πλεονεκτήματα που κατάφερε να επιτύχει η προτεινόμενη μέθοδος είναι το χαμηλό υπολογιστικό κόστος που απαιτεί, σε συνδυασμό με υψηλές ακρίβειες, καθιστώντας την ως μια αποδοτική λύση για διάφορες εφαρμογές. Τέλος, συμπεραίνεται ότι η μέθοδος προσφέρει σε μεγάλο βαθμό βελτιωμένη ερμηνευσιμότητα των χαρακτηριστικών, παρέχοντας βαθύτερη κατανόηση των αλληλεπιδράσεων και της σημασίας τους μέσω των χαρτών βαρών προσοχής.

6.2 Μελλοντικές Κατευθύνσεις

Υπάρχουν αρκετοί τομείς στους οποίους μπορεί να γίνει περαιτέρω έρευνα με την προτεινόμενη μέθοδο. Μια πρώτη πιθανή κατεύθυνση είναι η επέκταση της έρευνας στις τεχνικές προσεκτικής χρήσης αξιποιώντας τον μηχανισμό προσοχής που πραγματοποιήθηκαν στο μοντέλο DINOv2, εστιάζοντας στην προσαρμογή και αξιολόγηση της στο μοντέλο MAE. Επιπλέον, να πραγματοποιηθεί έρευνα για άλλες αυτο-επιβλεπόμενες

μεθόδους που εξάγουν χαρακτηριστικά είτε με την χρήση της μονάδας ταξινόμησης είτε χωρίς και η σύγκριση τους με την προτεινόμενη μέθοδο. Μία ακόμα πιθανή μελλοντική κατεύθυνση θα μπορούσε να είναι η εφαρμογή της προτεινόμενης μεθόδου σε διαφορετικό σετ δεδομένων, με στόχο τη σύγκριση της με βασικές μεθόδους αναφοράς (baselines). Τα εντυπωσιακά αποτελέσματα και η χαμηλή υπολογιστική απαίτηση της μεθόδου, υποδεικνύουν ότι μπορεί να εφαρμοστεί με επιτυχία και σε άλλες κατάντη εργασίες όπως η ανάλυση και τμηματοποίηση εικόνων και η αναγνώριση αντικειμένων, διευρύνοντας έτσι το πεδίο εφαρμογής της διάφορες εφαρμογές στην όραση υπολογιστών.

Βιβλιογραφία

[1] Jia Deng et al. «ImageNet: A large-scale hierarchical image database». In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 2009, pp. 248–255. DOI: 10.1109/CVPR.2009.5206848.

[2] Dan Hendrycks and Kevin Gimpel. «Gaussian Error Linear Units (GELUs)». In: *arXiv preprint arXiv:1606.08415* (2016).

[3] Mehdi Noroozi and Paolo Favaro. «Unsupervised Learning of Visual Representations by Solving Jigsaw Puzzles». In: *ECCV*. 2016.

[4] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. 2017. arXiv: 1412.6980 [cs.CV].

[5] Ashish Vaswani et al. «Attention is All you Need». In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[6] Yang You, Igor Gitman, and Boris Ginsburg. «Scaling SGD Batch Size to 32K for ImageNet Training». In: *CoRR* abs/1708.03888 (2017). arXiv: 1708.03888. URL: http://arxiv.org/abs/1708.03888.

[7] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. «Unsupervised Representation Learning by Predicting Image Rotations». In: *CoRR* abs/1803.07728 (2018). arXiv: 1803.07728. URL: http://arxiv.org/abs/1803.07728.

[8] Mehdi Noroozi et al. «Boosting Self-Supervised Learning via Knowledge Transfer». In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018, pp. 9359–9367. DOI: 10.1109/CVPR.2018.00975.

[9] Sanghyun Woo et al. «CBAM: Convolutional Block Attention Module». In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Sept. 2018.

[10] Ting Chen et al. «A Simple Framework for Contrastive Learning of Visual Representations». In: *Proceedings of the 37th International Conference on Machine Learning*. Ed. by Hal Daumé III and Aarti Singh. Vol. 119. Proceedings of Machine Learning Research. PMLR, 13–18 Jul 2020, pp. 1597–1607. URL: https://proceedings.mlr.press/v119/chen20j.html.

[11] Mathilde Caron et al. «Emerging Properties in Self-Supervised Vision Transformers». In: *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 9630–9640. DOI: 10.1109/ICCV48922.2021.00951.

[12] Alexey Dosovitskiy et al. «An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale». In: *International Conference on Learning Representations*. 2021. URL: https://openreview.net/forum?id=YicbFdNTTy.

[13] Alec Radford et al. «Learning Transferable Visual Models From Natural Language Supervision». In: *Proceedings of the 38th International Conference on Machine Learning*. Ed. by Marina Meila and Tong Zhang. Vol. 139. Proceedings of Machine Learning Research. PMLR, 18–24 Jul 2021, pp. 8748–8763. URL: https://proceedings.mlr.press/v139/radford21a.html.

[14] Hugo Touvron et al. «Going deeper with Image Transformers». In: *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 32–42. DOI: 10.1109/ICCV48922.2021.00010.

[15] Kaiming He et al. «Masked Autoencoders Are Scalable Vision Learners». In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 15979–15988. DOI: 10.1109/CVPR52688.2022.01553.

[16] Ioannis Kakogeorgiou et al. «What to Hide from Your Students: Attention-Guided Masked Image Modeling». In: *Computer Vision – ECCV 2022*. Springer Nature Switzerland, 2022, pp. 300–318. ISBN: 978-3-031-20056-4. DOI: 10.1007/978-3-031-20056-4_18. URL: https://link.springer.com/chapter/10.1007/978-3-031-20056-4_18.

[17] Jiahui Yu et al. «CoCa: Contrastive Captioners are Image-Text Foundation Models». In: *Transactions on Machine Learning Research* (2022). ISSN: 2835-8856. URL: https://openreview.net/forum?id=Ee277P3AYC.

[18] Jinghao Zhou et al. «iBOT: Image BERT Pre-Training with Online Tokenizer». In: *International Conference on Learning Representations (ICLR)* (2022).

[19] Maxime Oquab et al. *DINOv2: Learning Robust Visual Features without Supervision*. 2023.

[20] Bill Psomas et al. «Keep It SimPool:Who Said Supervised Transformers Suffer from Attention Deficit?» In: *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2023, pp. 5327–5337. DOI: 10.1109/ICCV51070.2023.00493.

[21] Xiaohua Zhai et al. «Sigmoid Loss for Language Image Pre-Training». In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. Oct. 2023, pp. 11975–11986.

[22] Adrien Bardes et al. «Revisiting Feature Prediction for Learning Visual Representations from Video». In: *arXiv:2404.08471* (2024).

[23] Alaaeldin El-Nouby et al. *Scalable Pre-training of Large Autoregressive Image Models*. 2024. URL: https://arxiv.org/abs/2401.08541.