

Όραση Υπολογιστών: Εργασία 4

Ζητείται να υλοποιηθεί αρχιτεκτονική συνελκτικού δικτύου σε Python με τη χρήση της βιβλιοθήκης [Keras-Tensorflow](#), που θα αφορά στο πρόβλημα της ταξινόμησης πολλαπλών κλάσεων (multi-class classification). Οι υποβολές των υλοποιήσεων θα γίνονται με τη μορφή Jupyter Notebook ενώ για την ανάπτυξη και δοκιμή θα χρησιμοποιηθεί η πλατφόρμα [Google Colab](#).

Στα πλαίσια της εργασίας θα πρέπει να υλοποιηθούν και να υποβληθούν ΔΥΟ αρχιτεκτονικές:

1. Ένα **ΜΗ προ-εκπαιδευμένο** δίκτυο το οποίο θα δημιουργηθεί αποκλειστικά για το τρέχον πρόβλημα ταξινόμησης, με τη διαδικασία που έχει αποτυπωθεί κατά το εργαστήριο.
2. Ένα **προ-εκπαιδευμένο** δίκτυο της επιλογής σας, που θα συνοδεύεται και από την αντίστοιχη ανάλυση του δικτύου που επιλέχθηκε.

Για το ερώτημα αυτό ΔΕΝ μπορείτε να χρησιμοποιήσετε το δίκτυο VGG-16 που αναλύθηκε στο εργαστήριο.

Να συμβουλευτείτε τον παρακάτω σύνδεσμο, όπου παρατίθεται η λίστα με τα προ-εκπαιδευμένα δίκτυα που είναι διαθέσιμα:

[Keras Applications](#)

Κάθε μια από τις προσεγγίσεις θα πρέπει να συνοδεύεται από:

Α) Πλήρη περιγραφή της αρχιτεκτονικής και των επιπέδων που χρησιμοποιούνται.

Β) Ποσοτική εκτίμηση της επίδοσής τους στο σύνολο δοκιμής, ως το ποσοστό επιτυχίας ταξινόμησης (accuracy).

Γ) Περιγραφή και δικαιολόγηση της διαδικασίας εκπαίδευσης (εποχές εκπαίδευσης, input size, batch size, callbacks, προ-επεξεργασία, data augmentation).

Δ) Δικαιολόγηση των επιλογών που έγιναν κατά τη δημιουργία της **ΜΗ προ-εκπαιδευμένης αρχιτεκτονικής** και τις ενδιάμεσες δοκιμές που οδήγησαν στις επιλογές αυτές.

Ιδιαίτερη έμφαση να δοθεί στην **επαύξηση του συνόλου δεδομένων** (βλ. Βοηθητικό υλικό). Να δικαιολογήσετε τις επιλογές μετασχηματισμών που επιλέξατε για το σκοπό αυτό, καθώς και να **παρουσιάσετε την πιθανή διαφορά στην απόδοση του δικτύου, με και χωρίς επαύξηση.**

Τέλος, **να δοκιμάσετε και να ποσοτικοποιήσετε την απόδοση (accuracy) των αρχιτεκτονικών δικτύου που δημιουργήσατε, στην βάση δεδομένων που σας δόθηκε στα πλαίσια της Εργασίας 3.**

Σημείωση 1: Στα πλαίσια της εργασίας αυτής δίνονται δύο βάσεις εικόνων ‘*imagedb*’ και ‘*imagedb_test*’. Η πρώτη θα χρησιμοποιηθεί για την εκπαίδευση του συστήματος και η δεύτερη για την δοκιμή και την αξιολόγησή του.

Οι βάσεις αυτές αποτελούν υποσύνολο της βάσης εικόνων [BelgiumTS - Belgian Traffic Sign Dataset \(ethz.ch\)](#).

Download link: <https://vc.ee.duth.gr:6960/index.php/s/wlnkxtlGmqBeATC>

Σημείωση 2: Δίνεται το αρχείο ‘*template.ipynb*’ το οποίο περιέχει κώδικα για απευθείας μεταφόρτωση της βάσης δεδομένων, χωρίς τη χρήση Google Drive.

Βοηθητικά σχόλια & Πηγές

Επαύξηση δεδομένων (Data Augmentation)

Συνήθως, τα βαθιά νευρωνικά δίκτυα επιτυγχάνουν καλύτερα αποτελέσματα όταν είναι διαθέσιμο μεγάλο πλήθος δεδομένων εκπαίδευσης. Είναι δυνατή η παραγωγή επιπλέον τεχνητών δεδομένων εκπαίδευσης από τις διαθέσιμες εικόνες, μέσω της εφαρμογής μετασχηματισμών. Κάποιοι μετασχηματισμοί που εφαρμόζονται συχνά είναι οι εξής:

- Flipping
- Random cropping
- Rotation
- Zoom

[tf.keras.preprocessing.image.ImageDataGenerator | TensorFlow Core v2.7.0](#)

[How to Configure Image Data Augmentation in Keras \(machinelearningmastery.com\)](#)

Χωρικές διαστάσεις

Οι αρχιτεκτονικές συνελκτικών νευρωνικών δικτύων που έχουμε δει στο εργαστήριο έχουν το εξής κοινό χαρακτηριστικό: Καθώς “βαθαίνει” το δίκτυο, οι χωρικές διαστάσεις των feature maps μειώνονται. Δύο τρόποι για να “διατηρήσουμε” τις χωρικές διαστάσεις των feature maps είναι οι εξής:

1. Αντί για το μοτίβο conv → pool → conv → pool, μπορούμε να εφαρμόσουμε πολλαπλά επίπεδα συνέλιξης πριν από κάποιο επίπεδο pooling (π.χ. conv → conv → conv → pool)
2. Χρήση padding “same” στα επίπεδα συνέλιξης
(βλ. [Conv2D layer \(keras.io\)](#))

Χρήση συνάρτησης flow_from_directory

Στην περίπτωση που χρησιμοποιηθεί η συνάρτηση flow_from_directory προσοχή στις παραμέτρους → (https://keras.io/preprocessing/image/#flow_from_directory)

Προσοχή: το target_size έχει default τιμή (256,256).

Batch size

Όσο μεγαλύτερο batch size, τόσο καλύτερα! Όμως, η μνήμη της κάρτας γραφικών δεν επαρκεί, έτσι επιλέγουμε το batch size στο μέγιστο δυνατό (μέχρι την τιμή εκείνη που εμφανίζεται memory error).

Επιπλέον υλικό

<https://keras.io/callbacks/>

[A Gentle Introduction to Batch Normalization for Deep Neural Networks \(machinelearningmastery.com\)](#)

[A Gentle Introduction to Dropout for Regularizing Deep Neural Networks \(machinelearningmastery.com\)](#)