



Άσκηση 5

Ανίχνευση Αντικειμένων (Object Detection)

Αντικείμενο - Στόχοι

- ✓ Εξοικείωση με αρχιτεκτονικές βαθιάς μάθησης για ανίχνευση αντικειμένων
- ✓ Αξιολόγηση των αποτελεσμάτων

Σε αυτή την άσκηση καλείστε να σχεδιάσετε και να εκπαιδεύσετε **δύο** μοντέλα ανίχνευσης αντικειμένων. Καλείστε να εργαστείτε σε ένα dataset από το Kaggle που μπορείτε να βρείτε [εδώ](#). Πρόκειται για ένα σετ δεδομένων που αφορά το πρόβλημα της ανίχνευσης αντικειμένων και περιέχει 6 κατηγορίες αντικειμένων (*dice, battery, toy car, spoon, highlighter, candle*) με εικόνες που προέκυψαν μετά από αναζήτηση στο Bing και από κάποια frames από Youtube videos. Συνολικά περιέχονται 1644 εικόνες με 2815 αντικείμενα προς ανίχνευση.

Καλείστε να σχεδιάσετε και να υλοποιήσετε 2 αλγορίθμους ανίχνευσης αντικειμένων (Μοντέλο Α και Μοντέλο Β) όπως περιγράφεται στα βήματα παρακάτω :

B1 : Προετοιμασία δεδομένων

- Κατεβάστε τα δεδομένα είτε από το helios.ntua.gr στη σελίδα του μαθήματος είτε από αυτό το [link](#)
- Σχεδιάστε και υλοποιήστε έναν αλγόριθμο τροφοδότησης δεδομένων για το πρόβλημα της ανίχνευσης αντικειμένων
- Προβείτε στους απαραίτητους μετασχηματισμούς και εφαρμόστε τουλάχιστον 2 γεωμετρικούς τρόπους επαύξησης των δεδομένων (data augmentation) για το μοντέλο Β.

B2 : Μοντέλο Α (Two-stage detector)

- Βασιστείτε σε ένα προεκπαιδευμένο δίκτυο τύπου Faster-RCNN από τη torchvision
- Εφαρμόστε μια διαδικασία εκπαίδευσης 4 βημάτων (όπως στην αρχική δημοσίευση του Faster-RCNN)
- Εφαρμόστε κριτήριο Early Stopping στην κατάλληλη μετρική

B3 : Μοντέλο Β (One-stage detector)

- Βασιστείτε σε ένα προεκπαιδευμένο δίκτυο τύπου SSD ή RetinaNet από τη torchvision
- Εκπαιδεύστε το δίκτυο στα δεδομένα σας
- Εφαρμόστε κριτήριο Early Stopping στην κατάλληλη μετρική

B4 : Αξιολόγηση και οπτικοποίηση αποτελεσμάτων

- Αξιολογήστε τα αποτελέσματα των μοντέλων σας στο σετ ελέγχου (test set) αξιοποιώντας τη μετρική mAP και άλλες σχετικές μετρικές
- Οπτικοποιήστε τα αποτελέσματα για 8 τυχαία επιλεγμένες εικόνες του σετ ελέγχου
- Σχολιάστε τα αποτελέσματα

Ζητούμενα

εκπονήστε τεχνική έκθεση περιγράφοντας τις διαδικασίες που ακολουθήσατε, απαντώντας και στα παρακάτω ερωτήματα

1. (50%) Εκτελέστε τα παραπάνω βήματα. Βασιστείτε στο [πρότυπο notebook](#) που υπάρχει στο github του μαθήματος

2. (50%) Δώστε απαντήσεις στα παρακάτω ερωτήματα:

1. Υποθέτουμε ότι κατά την εκπαίδευση του αλγορίθμου Faster-RCNN στο 1ο από τα 4 βήματα εκπαίδευσης επιλέγετε να εφαρμόσετε το κριτήριο πρόωρης διακοπής βασισμένο στη μετρική mAP για να διακόψετε τη διαδικασία εκπαίδευσης στο βέλτιστο αριθμό εποχών. Είναι αυτός ο τρόπος μια κατάλληλη εφαρμογή του κριτηρίου πρόωρης

διακοπής; Αιτιολογήστε και προτείνετε έναν εναλλακτικό τρόπο του κριτηρίου διακοπής για το συγκεκριμένο βήμα εκπαίδευσης.

2. Έχετε εκπαιδεύσει πλήρως ένα μοντέλο Faster-RCNN. Μελετώντας τα αποτελέσματα προκύπτει ότι ενώ τα αντικείμενα ανιχνεύονται ορθά και στις σωστές κατηγορίες, παρατηρούμε ότι τα bounding boxes που προκύπτουν δεν περιγράφουν κατά το βέλτιστο δυνατό τρόπο τα αντικείμενα. Τι θα μπορούσατε να βελτιώσετε κατά τη διαδικασία εκπαίδευσης ώστε να επιτύχετε μια καλύτερη προσαρμογή των bounding boxes των αντικειμένων στην εικόνα;
3. Ας υποθέσουμε ότι έχουμε συλλέξει δεδομένα από ένα UAV πάνω από έναν αυτοκινητόδρομο με στόχο να ανιχνεύσουμε τα αυτοκίνητα που διέρχονται. Κάθε εικόνα είναι ανάλυσης 4K, δηλαδή ανάλυσης 3840 x 2160 pixel και απεικονίζει ένα τμήμα αυτοκινητόδρομου μήκους περίπου 100 μέτρων. Τι δυσκολίες αναμένετε να αντιμετωπίσετε για το πρόβλημα της ανίχνευσης των οχημάτων; Προτείνετε πιθανές λύσεις για το πρόβλημα.
4. Ας υποθέσουμε ότι στα πλαίσια ανάπτυξης ενός αλγορίθμου καταμέτρησης πλήθους ανθρώπων σε μία συναυλία επιλέγουμε να σχεδιάσουμε έναν αλγόριθμο ανίχνευσης ως ενδιάμεσο βήμα. Ποιο υποτιμήμα ενός τυπικού αλγορίθμου ανίχνευσης θεωρείτε ότι θα προκαλέσει τις μεγαλύτερες δυσκολίες και γιατί;

Σημειώσεις

- Μαζί με την τεχνική έκθεση (pdf) να παραδώσετε και τον πηγαίο κώδικα που αναπτύξατε.
- Συστήνεται η χρήση της Pytorch και της Pytorch Lightning. Επιτρέπεται επίσης Keras/Tensorflow, JAX ή CNTK. Αν επιθυμείτε να εκπονήσετε την εργασία σε κάποια άλλη βιβλιοθήκη επικοινωνήστε πρώτα με τους διδάσκοντες του μαθήματος. Σε περίπτωση που δεν επιλέξετε την Pytorch, καλείστε να “μεταφέρετε” στην βιβλιοθήκη της επιλογής σας και το τμήμα της άσκησης που είναι ήδη υλοποιημένο στο πρότυπο notebook.

Χρήσιμα links

- <https://pytorch.org/vision/stable/models.html#object-detection-instance-segmentation-and-person-keypoint-detection>
- <https://pytorch.org/vision/stable/utils.html>
- <https://pytorch.org/vision/stable/io.html#image>
- https://torchmetrics.readthedocs.io/en/stable/detection/mean_average_precision.html
- https://alumentations.ai/docs/getting_started/bounding_boxes_augmentation/
- https://alumentations.ai/docs/getting_started/simultaneous_augmentation/

Σημείωση:

- Μέχρι και 17/05/2022 υπάρχει γνωστό bug στην MeanAveragePrecision μετρική της torchmetrics σχετικά με το device placement κάποιων τανυστών. Αν συναντήσετε πρόβλημα μπορείτε να το λύσετε απλά καλώντας “.cuda()” στο αντικείμενο της μετρικής στην αρχή κάθε εποχής validation (hook: `on_validation_epoch_start(self)`)!
 - Το .cuda() επιστρέφει την μετρική στην GPU

