

YOLOV5 NEIRONU TĪKLA APMĀCĪBA BUMBIERU ATPAZĪŠANAI DĀRZĀ

TRAINING OF YOLOV5 NEURAL NETWORK FOR PEAR DETECTION IN ORCHARD

Autori: **Artis FRIBERGS**, e-pasts: af21042@edu.rta.lv,
Edmunds LUKAŠEVIČS, e-pasts: el21105@edu.rta.lv,
Guntis LIELBĀRDIS, e-pasts: gl21049@edu.rta.lv

Zinātniskā darba vadītājs: **Sergejs KODORS**, Dr.sc.ing, e-pasts: sergejs.kodors@rta.lv
Rēzeknes Tehnoloģiju akadēmija, Atbrīvošanas aleja 115, Rēzekne

Abstract. *A fruit-growing is an important branch of agriculture for various reasons. Fruits provide essential nutrients and vitamins to our diet, and they are also a significant source of income for fruit-growers. To improve the efficiency of fruit cultivation, we trained a pear detection neural network with YOLOv5 architecture using a dataset from the project lzp-2021/1-0134. The dataset contained 1273 photographs of pear trees with image sizes 640x640px. We had trained the neural network model YOLOv5m five times and achieved the best result equal to mAP@0.5 0.8 and mAP@0.5:0.95 0.43. The use of artificial intelligence in fruit cultivation can help to optimize the planning of fruit picking, contributing to the precision horticulture.*

Keywords: *artificial intelligence, fruits, object detection, precision horticulture, YOLOv5.*

Ievads

Augļkopība ir svarīga daudzu iemeslu dēļ. Augļi ir veselīgi un barojoši pārtikas produkti, kuri nodrošina mūsu organismam nepieciešamus vitamīnus un minerālvielas. Turklāt, augļi ir svarīgs ekonomikas un tirdzniecības nozares produkts, nodrošinot darbavietas miljoniem cilvēku visā pasaulē. Augļi ir arī iecienīts un populārs pārtikas produkts, ko patērē cilvēki visā pasaulē, tādējādi veicinot tirdzniecību un ekonomisko izaugsmi.

Bumbieru ražas prognozēšana lielos komerciālos dārzos varbūt sarežģīts un laikietilpīgs process, jo prasa precīzu un detalizētu datu vākšanu, analīzi un interpretāciju. Mākslīgā intelekta (MI) izmantošana var ievērojami vienkāršot šo procesu, jo pielietojot MI var automātiski apsekot dārzu un saskaitīt augļu daudzumu konkrētās dārza rindās un kvartālos. Tādējādi, dārzkopji var iegūt precīzus datus par bumbieru ražu, kas palīdzēs optimizēt augļkopības procesus un veicināt efektīvu resursu izmantošanu. Turklāt, MI izmantošana var būt arī izdevīga no ekonomiskā viedokļa, jo ietaupīts laiks un resursi var būt novirzīti citiem uzdevumiem, kas var paaugstināt augļkopības procesa efektivitāti un rentabilitāti.

Neironu tīkli ir viena no aktuālākajām un strauji attīstītajām tehnoloģijām mūsdienu datorzinātnē, kas spēj apstrādāt un interpretēt datus daudzās jomās, tostarp viedā lauksaimniecībā. Šajā rakstā tiek apskatīta neironu tīklu pielietošana bumbieru saskaitīšanai dārzos.

Pētījuma mērķis: apmācīt YOLOv5m neironu tīkla modeli, kas spēj meklēt bumbierus dārza fotogrāfijās.

Uzdevumi:

- 1) sagatavot datu kopu neironu tīkla apmācīšanai;
- 2) apmācīt neironu tīklu, pielietojot YOLOv5;
- 3) novērtēt apmācītu neironu tīklu bumbieru atpazīšanas kvalitāti.

Materiāli un metodes

Lai apmācītu *YOLOv5* neironu tīklu, tika izmantota datu kopa no projekta *lzp-2021/1-0134*. Datu kopa sastāv no 1273 attēliem ar izmēru 640x640 punkti, kuros redzami koki ar bumbieriem. Anotāciju veikšanai tika izmantotas robežkastes (*bounding boxes*). Attēli un *YOLOv5* anotācijas faili tika augšupielādēti vienā mapē, lai tos varētu nejauši sakārtot, izmantojot *Python* skriptu, trijās mapēs. Apmācībai ir vajadzīgas divas mapes: trenēšanas un validācijas kopas; un viena mape – neironu tīkla testēšanai. Piemērs ar diviem attēliem no datu kopas ir redzams zemāk (skat. 1. att.).



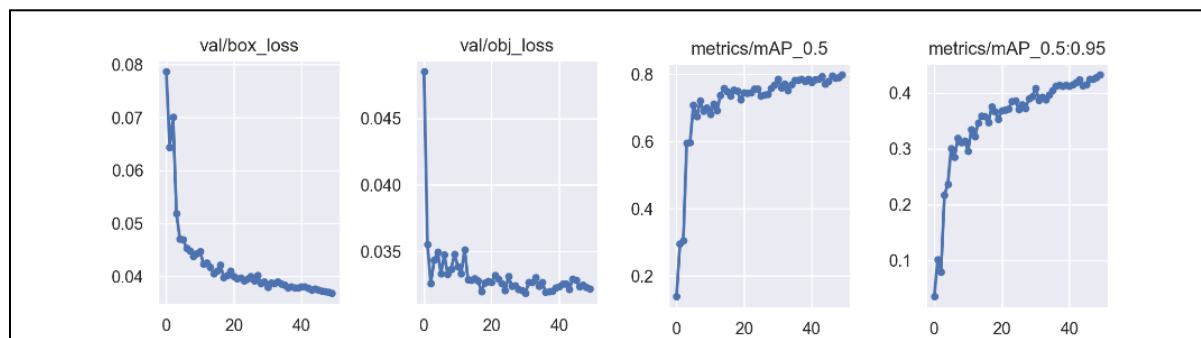
1. attēls. Attēlu piemēri no datu kopas

Neironu tīklu apmācīšanai tika izmantoti šādi rīki un tehnoloģijas – *MakeSense* [1], *YOLOv5* [2], *Python*, dators ar *NVIDIA* videokarti, kas atbalstā *CUDA* [3] tehnoloģiju. Iepriekš sagatavota datu kopa tika nejauši sadalīta trijās mapēs ar attiecību 70%, 20% un 10%. Sadalījums nepieciešams, lai apmācītu un pēc tam pārbaudītu modeļa atpazīšanas kvalitāti.

Nākamais solis bija apmācīt neironu tīklu ar *YOLOv5m* modeli, izmantojot personīgo datoru un pielietojot *GPU* (*Graphics Processing Unit*) skaitļošanu. Lai apmācītu *YOLOv5* neironu tīklu, tika veiktas šādas darbības: instalētas *YOLOv5* pakotnes (atkarības), importēta datu kopa, definēta modeļa konfigurācija un arhitektūra, apmācīts *YOLOv5m* modelis, novērtēta *YOLOv5m* veiktspēja, vizualizēti neironu tīkla treniņu dati, pārbaudīti treniņi ar testa attēliem un eksportēti rezultāti turpmākiem secinājumiem. Lai iegūtu precīzākus treniņa rezultātus, visas šīs darbības tika veiktas 5 reizes, katru reizi izmainot trenēšanas, validācijas un pārbaudes kopas.

Rezultāti un to izvērtējums

Lai nodrošinātu precizitāti, neironu tīklu apmācību veica piecas reizes, lai varētu vēlāk novērtēt tās efektivitāti un izvēlēties labāku rezultātu. Pēc katras apmācības mēs saņēmām grafikus ar atšķirīgiem rezultātiem. Lai turpmāk veiktu aprēķinus, mēs saglabājam testa grafikus, jo tie satur nepieciešamo informāciju, lai novērtētu cik labi tika pamācīts neironu tīkls. Zemāk ir parādīti labākie apmācību rezultāti (skat. 2. att.).



2. attēls. Apmācības rezultāti: labākais gadījums

Attēlā ir četri grafiki ar zaudējuma un precizitātes funkcijām.

Zaudējuma funkcijas:

- **val/ box_loss** – kļūda, ka lauka prognoze nosedz objektu neprecīzi;
- **val/ obj_loss** – kļūda no nepareizas *IoU* (*Intersection over Union*) prognozes priekš robežkastes. *IoU* mēra pārklāšanu starp divām robežām. Tas tiek izmantots, lai izmērītu, cik stipri atrasta objekta robežas pārklājas ar gaidītajām robežām.

Precizitātes funkcijas:

- ***mAP*** (*mean Average Precision*) salīdzina pamata patiesības robežkastes ar noteiktu lauku un aprēķina vidējo rezultātu. Jo augstāks rādītājs, jo precīzāks ir modeļa noteikšanas spēja.
- ***mAP@0.5*** - ja *IoU* ir iestatīts uz 0,5, tiek aprēķināta vidēja precizitāte visiem attēliem, un tad tiek aprēķināts visu kategoriju vidējais rādītājs (*mAP*). Vidējā precizitāte ir populāra metrika objektu detektoru precizitātes mērīšanai. Tas aprēķina vidējo precizitātes vērtību diapazonā no 0,0 līdz 1,0;
- ***mAP@0.5:0.95*** – izteic vidējo *mAP* pie dažādām *IoU* vērtībām (no 0,5 līdz 0,95 ar soli 0,05) [4].

Lai saprastu, cik labi modelis atpazīst objektus, un piefiksētu precizitāti, pētījumos izmantosim *mAP@ 0.5:0.95* maksimālās vērtības. Zemāk var redzēt tabulu ar *mAP@0.5:0.95* maksimālām vērtībām no visām 5 apmācībām (skat. 1. tab.).

1. tabula

***mAP@0.5:0.95* maksimālās vērtības pēc katras apmācības**

Nr	1	2	3	4	5
<i>mAP@0.5:0.95</i>	0,39382	0,41422	0,41546	0,41949	0,43243

Izmantojot vērtības no tabulām, mēs noteiksim minimālo vērtību, maksimālo vērtību, vidējo vērtību un mediānu, lai izteiktu izvēlēta risinājuma (*YOLOv5m*) kvalitāti.

Lai būtu vieglāk atrasts minimālo vērtību, maksimālo vērtību un mediānu, vērtības 1. tabulā bija sakārtotas augošā secībā.

Pēc vērtību sakārtošanas varam secināt, ka:

- Minimālā *mAP@ 0.5:0.95* vērtība – 0,39382;
- Maksimālā *mAP@ 0.5:0.95* vērtība – 0,43243.

Mediāna tiek aprēķināta pielietojot nepāra skaitļa formulu (1), iegūstot rezultātā vērtību – 0,41546.

$$Me = X \left[\frac{n+1}{2} \right], \quad (1)$$

kur *X* – sakārtots vērtību saraksts; *n* – vērtību skaits datu kopā.

Lai atrastu vidējo vērtību izmantojam nepieciešamo formulu (2) un iegūstam rezultātu – 0,415084.

$$V = \frac{x_1+x_2+\dots+x_n}{n}, \quad (2)$$

kur *x* – vērtība no kopas *X*, *n* – vērtību skaits kopā *X*.

Beigās, lai pārbaudītu cik labi strādā apmācītais neironu tīkls, tika pielietota vizuāla metode. Pielietojot apmācīto neironu tīklu, tika apstrādātas bildes. Pārskatot attēlus, mēs pamanījām, ka dažos attēlos neironu tīkls atzīmēja bumbierus kur droši nevar pateikt ka tur ir bumbieris, ka arī ne ļoti precīzi uzzīmētas robežas. Piemēru ar šādiem attēliem var apskatīt zemāk (skat. 3. att.):



3. attēls. Bumbieru atpazīšanas piemērs, pielietojot apmācīto neironu tīklu

Secinājumi

Izmantojot *MakeSense* rīku, personīgo datoru ar *NVIDIA* videokarti, kura atbalsta *CUDA* tehnoloģiju, tika apmācīts *YOLOv5m* neironu tīkls, kas spēj meklēt bumbierus dārza fotogrāfijās. Eksperimentā tika izmantota datu kopa no projekta *lzp-2021/1-0134* ar 1273 attēliem.

Katras apmācības beigās saņemam grafikus ar zaudējuma un precizitātes funkcijām, bet pētījumā, lai fiksētu modeļa precizitāti, izmantojam *mAP@0.5:0.95* maksimālo vērtību. Pētījuma rezultātā tika iegūtas 5 dažādas vērtības, kur minimālā vērtība bija 0,39382 un maksimālā vērtība bija 0,43243. Tāpat, izmantojot formulas, tika atrasta mediāna – 0,41546 un vidējā vērtība – 0,415084. Izmantojot *YOLOv5* iebūvētu skriptu neironu tīkla pārbaudei un testa attēlus, tika pārbaudīta labākā neironu tīkla noteikšanas precizitāte, kur neironu tīkls mēģināja atrast bumbierus uz bumbieru koka. Izpētot šos attēlus, mēs secinājām, ka dažos attēlos neironu tīkls atrada bumbierus, kur to nav, dažās – atrada, bet ne ļoti precīzi iezīmēja bumbieru robežas. Bet tomēr vairumam atrada un iezīmēja diezgan precīzi, kas protams ir subjektīvs viedoklis.

Vērts pieminēt, ka modeļa precizitāti var uzlabot dažādos veidos. Viens no tiem, atbildīgāki atteikties pret robežkastu zīmēšanu. Arī ir vērts palielināt datu kopas apjomu un fotogrāfijas kvalitāti. Protams tās viss palielinās gan cilvēkresursus gan skaitļošanas jaudu.

Summary

Growing fruits is crucial to the world's food supply and overall health. Fruits provide essential vitamins, minerals, and fiber that are necessary for a healthy diet. In addition, fruits are a significant source of income and employment for many communities around the world. Forecasting the yield of fruit crops, such as pears, is a challenging task due to the large scale of their cultivation and the complex interactions between various environmental factors.

Using neural networks to detect fruits on trees involves training the network with images of trees and their fruits. The network can then use computer vision techniques to recognize and count fruits based on their shape, size, color, and texture. The use of artificial

intelligence can be particularly useful for large-scale orchards, where manual counting of fruits on trees can be time-consuming and labor-intensive. In this article, we explored the possibility of using YOLOv5 neural network to detect pears in orchards.

A neural network, YOLOv5m, has been trained on a dataset from the project lzp-2021/1-0134. The dataset contained 1273 images. In these photographs, we marked each pear using the MakeSense image annotation tool. The training was conducted on PC with an NVIDIA graphics card that supports CUDA technology. This approach allowed us to train the neural network to identify pears in images with average accuracy, which can be useful for large orchards to complete yield forecasting. The use of machine learning technologies like YOLOv5 is a significant step forward in the automation of agricultural processes and can greatly benefit the industry by improving efficiency and productivity.

After each training, we obtained graphs with loss and accuracy functions, but in the research, we use the maximum values of mAP@0.5: 0.95, because this value helps to capture the accuracy of the model. As a result of the research, 5 different values were obtained, where the minimum value was 0.39382 and the maximum value was 0.43243. Also, using the formulas, a median of 0.41546 and mean of 0.415084 were found.

While analyzing the test results, it was discovered that in some images, the neural network detected pears where they were not presented, while in others, it detected them but did not mark their boundaries correctly. However, for the majority of the images, it detected and marked the pears with a relatively high level of accuracy.

Acknowledgment

This research is funded by the Latvian Council of Science, project “Development of autonomous unmanned aerial vehicles based decision-making system for smart fruit growing”, project No. lzp2021/1-0134.

Literatūras un avotu saraksts

1. MakeSense. <https://www.makesense.ai>, sk. 02.04.2023.
2. Ultralytics. How to train Custom Data with YoLoV5. <https://github.com/ultralytics/yolov5/wiki/Train-Custom-Data>, sk. 02.04.2023.
3. NVIDIA. CUDA. <https://developer.nvidia.com/cuda-toolkit>, sk. 02.04.2023
4. Eléa Petton. Object detection: train YOLOv5 on a custom dataset. <https://blog.ovhcloud.com/object-detection-train-yolov5-on-a-custom-dataset/>, sk. 02.04.2023