

CHORRAHADA TRANSPORT VOSITALARI SONINI ANIQLASHDA YOLOV8 PYTHON ALGORITMINI ISHLAB CHIQISH

Isoqov Fayozbek Faxriddinovich ¹, Xaydarov Sherhali Islom o'g'li ²

¹ Denov tadbirkorlik va pedagogika instituti tyutori

² Denov tadbirkorlik va pedagogika instituti doktoranti

ARTICLE INFO.

Kalit so'zlar: YOLOv8, bashorat qilish, model, ultralytics, obyekt.

Annotatsiya

Obyektlarni aniqlash uchun ma'lumotlarni belgilovchi va 'yolov8n.pt' formatda saqlashni ta'minlovchi ma'lumotlardir. YOLOv8 algoritmi yoki boshqa obyekt TANIMA algoritmlari Chorrahada transport vositalari sonini aniqlashda keng foydalaniladi. Obyektlarning joylashuvi (bounding box coordinates) va ularning turini (class label) ko'rsatadi. Odatda '.txt', '.xml', '.json' kabi formatlarda bo'lishi mumkin, ammo YOLOv8 uchun ko'p o'zgaruvchanliklar va uslubga ega va samarali foydalanish imkoniyatiga ega hisoblanadi.

<http://www.gospodarkainnowacje.pl/> © 2023 LWAB.

Kirish

YOLOv8 algoritmi uchun datani dasturlash jarayonida, yuqoridagi ko'rinishda annotatsiyalarni yozishingiz lozim. Ma'lumotlar o'rganish jarayonida, har bir obyekt uchun to'g'ri tur va bounding box koordinatalarini to'g'ri ko'rsatish muhimdir, chunki modelni o'rganish uchun aniq ma'lumotlar kerak bo'ladi.

YOLOv8 - bir bosqichli ob'ektni aniqlash modeli bo'lib, u tasvirdagi ob'ektlar uchun chegaralovchi qatlamlar va sinf ehtimolini bashorat qilish uchun yagona konvolyutsion neyron tarmog'idan (CNN) foydalanadi. Bu YOLO ob'ekt detektorlari oilasining eng so'nggi versiyasi bo'lib, u turli ko'rsatkichlar bo'yicha eng zamonaviy ko'rsatkichlarga erishishi ko'rsatilgan.

YOLOv8 Python Ushbu algoritm yordamida ob'ektni aniqlash, segmentlash va tasniflash uchun YOLOv8-ni Python loyihalarigizga muammosiz integratsiya qilishda yordam berish uchun mo'ljallangan. Foydalanish uchun qulay Python interfeysi YOLOv8 ni Python loyihalariga kiritmoqchi bo'lgan har bir kishi uchun qimmatli manba bo'lib, ilg'or ob'ektlarni aniqlash imkoniyatlarini tezda amalga oshirish imkonini beradi.

Misol uchun, foydalanuvchilar modelni yuklashlari, uni o'qitishlari, tekshirish to'plamida ishlashini baholashlari va hatto bir necha qator kodlar bilan ONNX(bu mashinani o'rganish modellarini ko'rsatish uchun yaratilgan ochiq format) formatiga eksport qilishlari mumkin.

```
from ultralytics import YOLO
```

```
# Noldan yangi YOLO modelini yarating
```

```

model = YOLO('yolov8n.yaml')
# Oldindan tayyorlangan YOLO modelini yuklang (trening uchun tavsiya etiladi)
model = YOLO('yolov8n.pt')
# 3 davr uchun "baza128.yaml" ma'lumotlar to'plamidan foydalanib modelni
natijani = model.train(data='baza128.yaml', epochs=3)
# Validatsiya to'plamida modelning ishlashini baholang
natija= model.val()
# Model yordamida tasvirda ob'ektni aniqlashni amalga oshiring
results = model(rasm.jpg')
# Modelni ONNX formatiga eksport qiling
muvaffaqiyat = model.export(format='onnx')

```

Oldindan tayyorlangan (tavsiya etiladi)

```

from ultralytics import YOLO
model = YOLO('yolov8n.pt')
# har qanday model turidan o'tish
natijalar = model.train(epochs=5)
from ultralytics import YOLO
model = YOLO('yolov8n.yaml')
natijalar = model.train(data='baza28.yaml', epochs=5)
model = YOLO("last.pt")
natijalar = model.train(resume=True)

```

Val rejimi YOLOv8 modelini o'qitgandan keyin tekshirish uchun ishlatiladi. Ushbu rejimda model uning aniqligi va umumlashtirish samaradorligini o'lchash uchun tekshirish to'plamida baholanadi. Ushbu rejimdan modelning ish faoliyatini yaxshilash uchun giperparametrlarini sozlash uchun foydalanish mumkin.

```

from ultralytics import YOLO
model = YOLO('yolov8n.yaml')
model.train(data='baza128.yaml', epochs=5)
model.val()
# U siz o'qigan ma'lumotlarni avtomatik ravishda baholaydi.
from ultralytics import YOLO
model = YOLO("model.pt")
# Agar siz ma'lumotlarni o'rnatmasangiz, u model.pt'dagi YAML ma'lumotlar faylidan
foydalanadi.
model.val()
# yoki siz o'zingiz xohlagan ma'lumotlarni o'rnatishingiz mumkin
model.val(data='baza128.yaml')

```

Bashorat qilish(Prediction)

Bashorat qilish rejimi yangi tasvirlar yoki videolarda o'qitilgan YOLOv8 modelidan foydalangan holda bashorat qilish uchun ishlatiladi. Ushbu rejimda model nazorat nuqtasi faylidan yuklanadi va foydalanuvchi xulosa chiqarish uchun rasm yoki videolarni taqdim etishi mumkin. Model kirish tasvirlari yoki videolaridagi ob'ektlarning sinflari va joylashishini bashorat qiladi.

```

from ultralytics import YOLO
from PIL import Image
import cv2
model = YOLO("model.pt")
# barcha formatlarni qabul qiladi - image/dir/Path/URL/video/PIL/ndarray. 0 for webcam
natijalar = model.predict(source="0")
natijalar = model.predict(source="folder", show=True)
# Predlarni ko'rsatish. Barcha YOLO bashoratli dalillarni qabul qiladi
# from PIL
im1 = Image.open("avtomobil.jpg")
results = model.predict(source=im1, save=True)
im2 = cv2.imread("avtomobil.jpg")
natijalar = model.predict(source=im2, save=True, save_txt=True)
natijalar = model.predict(source=[im1, im2])

```

Natijalardan foydalanish

```

results = model.predict(source="folder")
results = model.predict(source=0, stream=True)
for result in results:
    result.boxes.xyxy # xyxy formatli quti, (N, 4)
    result.boxes.xywh # xywh formatli quti, (N, 4)
    result.boxes.xyxy # xyxy formatli quti, lekin normallashtirilgan, (N, 4)
    result.boxes.xywhn # xywh formatli quti, lekin normallashtirilgan, (N, 4)
    result.boxes.conf # ishonch balli, (N, 1)
    result.boxes.cls # cls, (N, 1) # Segmentatsiya
    result.masks.data # niqoblar, (N, H, W)
    result.masks.xy # x,y segmentlari (piksellar), List[segment] * N
    result.masks.xyn # x,y segmentlari (normallashtirilgan), List[segment] * N
    result.probs # cls prob, (num_class, )
    result = result.cuda()
    result = result.cpu()
    result = result.to("cpu")
    result = result.numpy()

```

Eksport(Export)

Eksport rejimi YOLOv8 modelini joylashtirish uchun ishlatilishi mumkin bo'lgan formatga eksport qilish uchun ishlatiladi. Ushbu rejimda model boshqa dasturiy ilovalar yoki apparat qurilmalari tomonidan ishlatilishi mumkin bo'lgan formatga aylantiriladi. Ushbu rejim modelni ishlab chiqarish muhitiga joylashtirishda foydalidir.

ONNX ga eksport qiling

```
from ultralytics import YOLO
model = YOLO('yolov8n.pt')
model.export(format='onnx', dynamic=True)
```

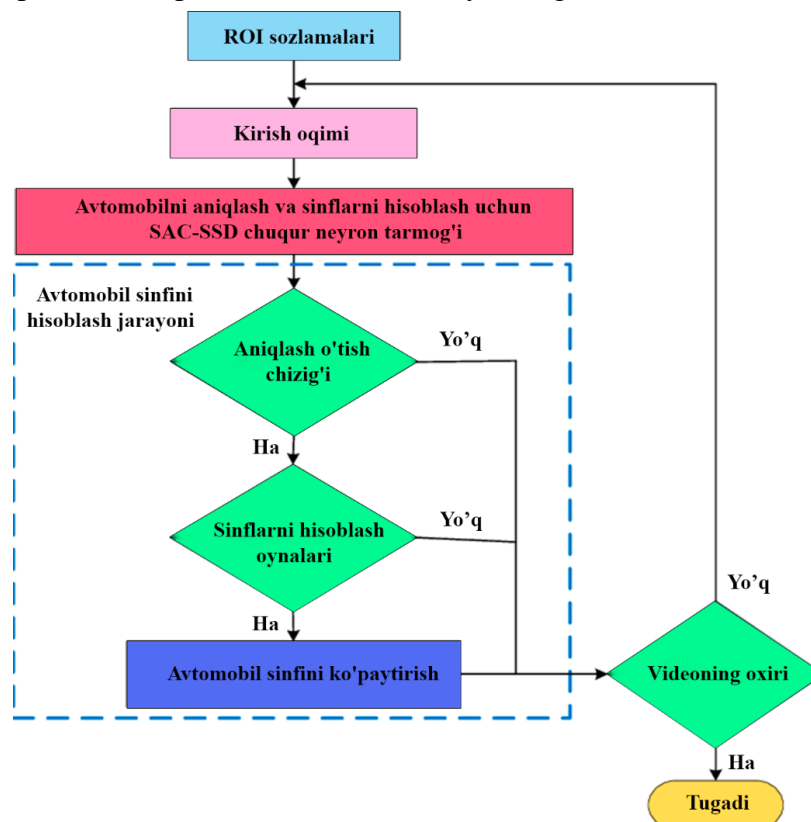
TensorRT ga eksport qiling

```
from ultralytics import YOLO
model = YOLO('yolov8n.pt')
model.export(format='onnx', device=0)
```

Trek(Track), Python

Track rejimi YOLOv8 modelidan foydalangan holda real vaqt rejimida ob'ektlarni kuzatish uchun ishlatiladi. Ushbu rejimda model nazorat punkti faylidan yuklanadi va foydalanuvchi real vaqtda ob'ektni kuzatishni amalga oshirish uchun jonli video oqimini taqdim etishi mumkin. Ushbu rejim kuzatuv tizimlari yoki o'zi boshqariladigan avtomobillar kabi ilovalar uchun foydalidir.

```
from ultralytics import YOLO
# Load a model
model = YOLO('yolov8n.pt') # Rasmiy aniqlash modelini yuklang
model = YOLO('yolov8n-seg.pt') # rasmiy segmentatsiya modelini yuklang
model = YOLO('path/to/best.pt') # maxsus modelni yuklang
```



Ushbu maqola YOLOv8 algoritmining nozik tomonlarini o'rganib chiqadi, uning umumiy dizayni, model tuzilishi, yo'qotish funktsiyasi, ma'lumotlarni oshirish usullari va xulosa chiqarish jarayonini har tomonlama o'rganishni taklif qiladi. Tushunishga yordam berish uchun ko'plab diagrammalar taqdim etiladi. Xulosa qilib aytganda, YOLOv8 tasvir tasnifi, Anchor-Free obyektini aniqlash va misol segmentatsiyasini o'z ichiga olgan yuqori samarali algoritmdir. Uning aniqlash komponenti yangi samaradorlik darajalariga erishish uchun ko'plab zamonaviy YOLO algoritmlarini o'z ichiga oladi.

Foydalanilgan adabiyotlar

1. Хайдаров, ш. (2023). Формирование базы данных предупреждений о дорожных знаках с использованием алгоритма распознавания объектов. *Предпринимательства и педагогика*, 5(1), 103-112.
2. Karimova, z., xaydarov, s., & doniyorova, g. (2023). Манипулирование изображениями с помощью языка программирования при распознавании дорожных знаков. *Предпринимательства и педагогика*, 5(1), 112-119.
3. Ziyatovich, m. F., & islom o'g'li, x. S. (2023). Sun'iy intellekt va uning ta'lim sohasiga alohida murojaat qilgan holda turli sohalardagi qamrovi. *Образование наука и инновационные идеи в мире*, 16(3), 16-19.
4. Мамажанов, р. Я., & хайдаров, ш. И. (2022). Разработка платформы инновационного подхода при подготовке научных статей на международном уровне. *Central asian journal of mathematical theory and computer sciences*, 3(12), 267-269.
5. Мамажанов, р., & хайдаров, ш. (2022). Цифровая экономика в повышении экономики и влияния вузов расположение в центре. *Central asian journal of mathematical theory and computer sciences*, 3(12), 270-275.
6. Мамажанов, р. Я., & хайдаров, ш. И. (2022). Разработка методов и алгоритмов для распознавание дорожных знаков. *Central asian journal of mathematical theory and computer sciences*, 3(10), 50-57.
7. Мамажанов, р. Я., & хайдаров, ш. И. (2022). Создания web приложения и распознавания ограничения скорости дорожных знаков. *Central asian journal of mathematical theory and computer sciences*, 3(4), 57-61.