



پروژه دوم درس مباحث ویژه

تشخیص حالات دست

استاد:

دکتر کیانی

تهیه کننده:

رضا اعلایی

فهرست

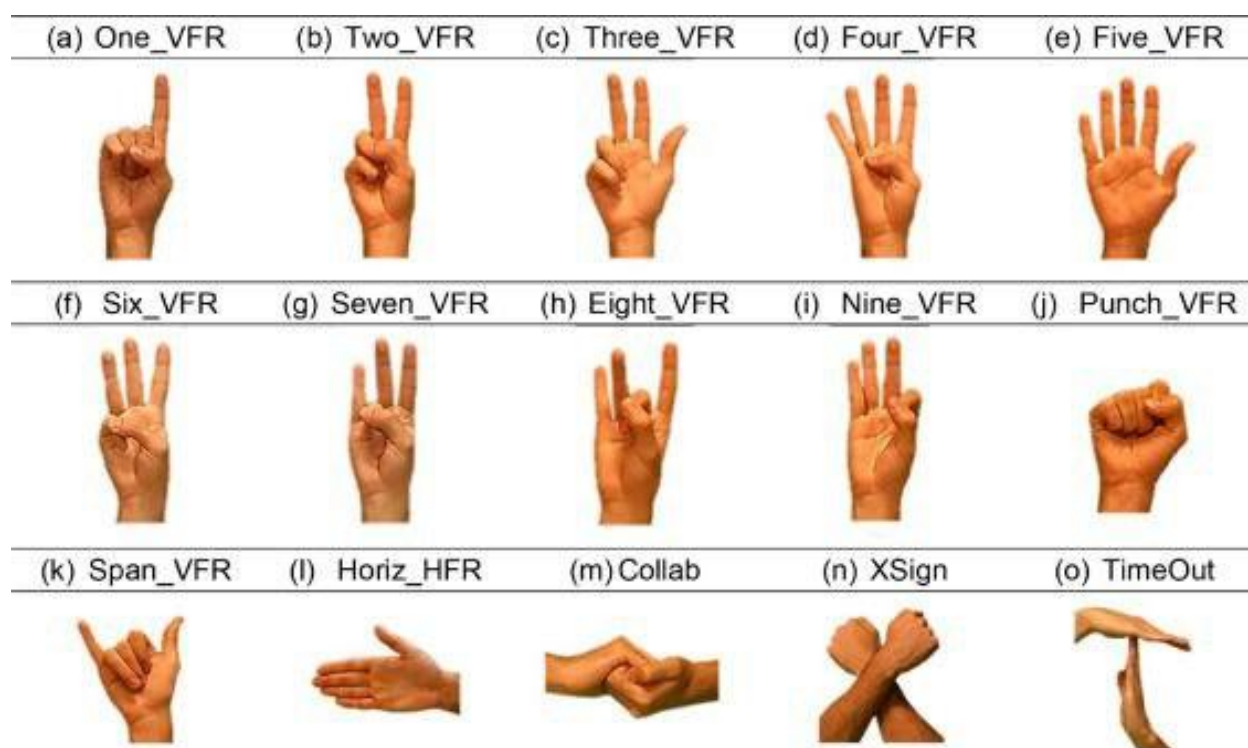
۲.....	شرح پروژه
۳.....	اقدامات کلی پروژه
۳.....	لیبل گذاری
۴.....	آموزش مدل نهایی
۵.....	گزارش نهایی مدل و تست آن

شرح پروژه

در این پروژه قصد داریم که حالت مختلف دست که در تصویر زیر آورده شده است را به کمک آموزش یک مدل YOLO پیاده‌سازی کنیم.

دیتاست به کار رفته در این پروژه، دیتاست sign language for numbers است که در kaggle موجود می‌باشد و حالات دست یک تا نه موجود در تصویر را پوشش می‌دهد (مدل ما نیز باید بتواند حالات یک تا نه را تشخیص دهد). مدل YOLO v8 نیز برای این کار استفاده شده است.

بستر اجرایی پروژه نیز google colab می‌باشد.



اقدامات کلی پروژه

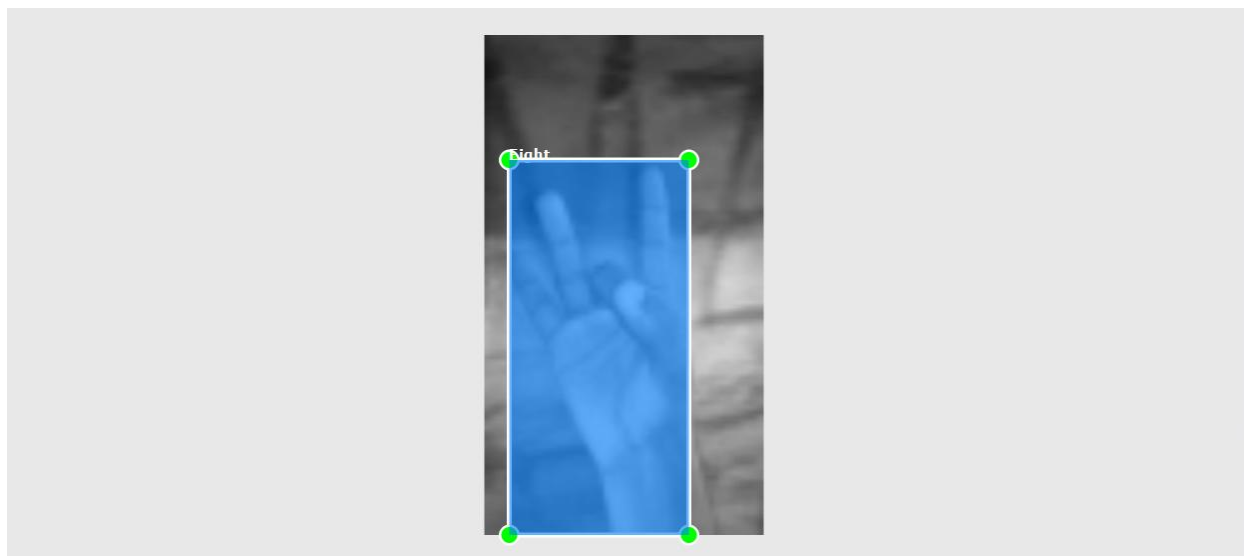
روند کلی این پروژه به شرح زیر است.

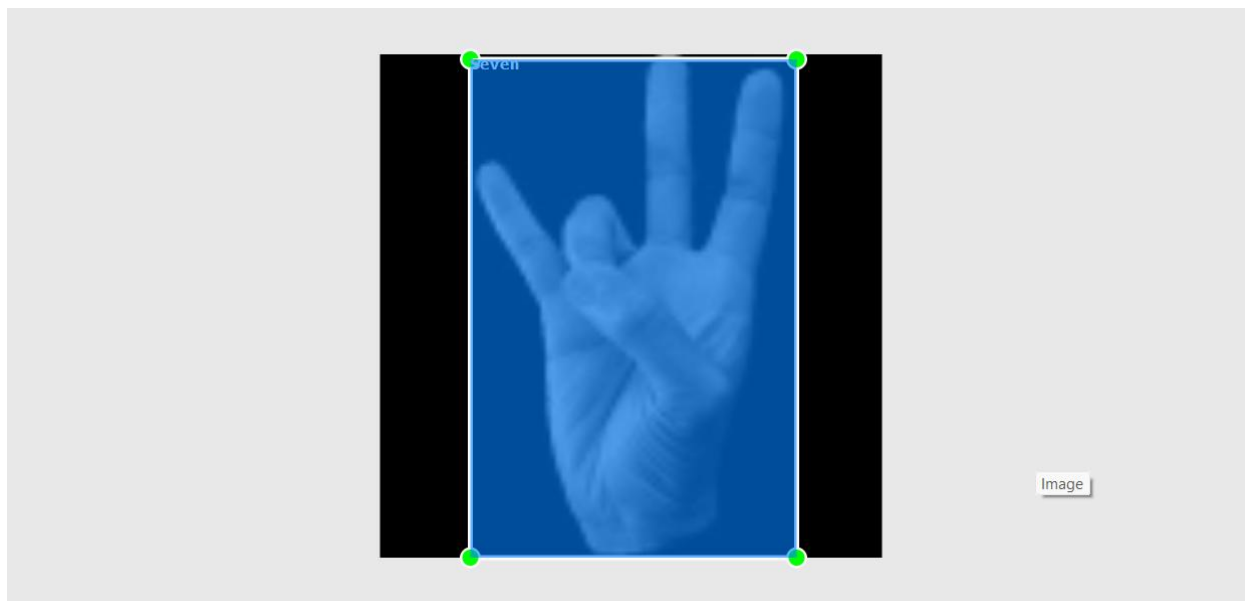
چالش اصلی این پروژه پیدا کردن دیتاست‌های مناسب برای آموزش مدل است. پس از بافتن این دیتاست‌ها باید عملیات لیبل گذاری برای تصاویر انجام شود که کاری زمانبر است.

پس از آن نیز باید دیتاهای لیبل شده به مدل بذای آموزش داده شوند.

لیبل گذاری

برای این کار از labelImg استفاده شده است که دو نمونه از موارد لیبل گذاری شده در زیر مشخص است.





آموزش مدل نهایی

برای آموزش نهایی مدل، ابتدا دیتاهای لیبیل شده را به دو دسته‌ی train و validation تقسیم میکنیم. فایل dataset.yaml را برای آن ساخته و شروع به آموزش مدل میکنیم. نتیجه‌ی اجراهای مدل و وزن‌ها و موارد مورد نیاز حاصل از آموزش مدل در فایل detect در گوگل درایو پیوست شده بارگذاری شده است.

گزارش نهایی مدل و تست آن

گزارش مدل به شرح کامل در فولدر detect در گوگل درایو موجود است. اینجا فقط بخش کوچکی از نتیجه مدل آورده شده است.

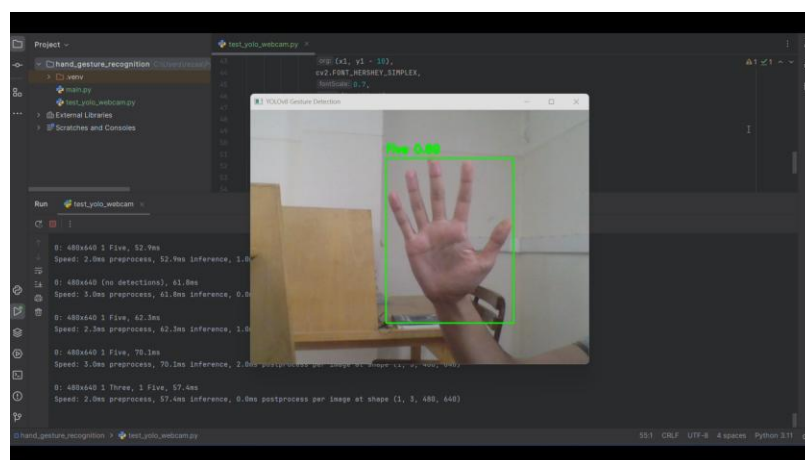
```
Validating runs/detect/train/weights/best.pt...
Ultralytics 8.3.55 Python-3.10.12 torch-2.5.1+cu121 CUDA:0 (Tesla T4, 15102MiB)
Model summary (fused): 168 layers, 3,007,403 parameters, 0 gradients, 8.1 GFLOPs
```

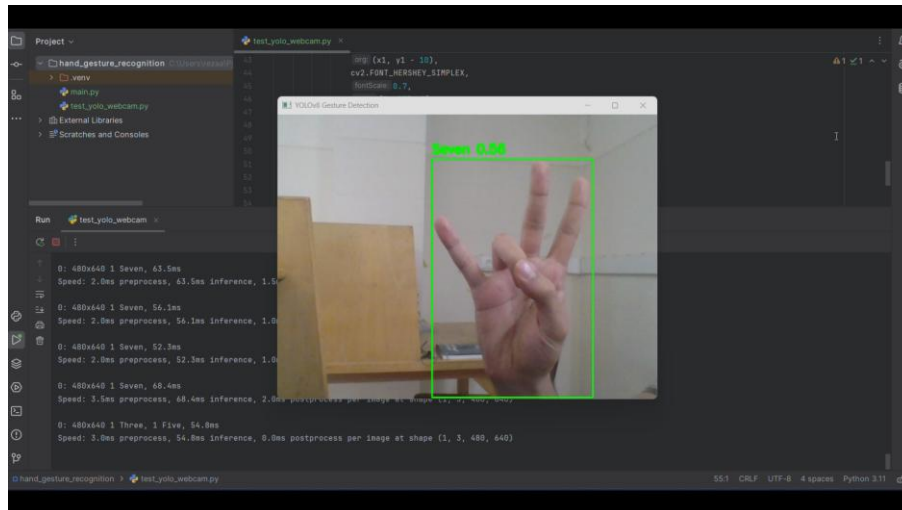
Class	Images	Instances	Box(P)	R	mAP50	mAP50-95): 100%
all	170	170	0.992	0.99	0.995	0.951
One	20	20	1	0.954	0.995	0.931
Two	16	16	0.975	1	0.995	0.954
Three	23	23	0.992	1	0.995	0.947
Four	20	20	0.993	1	0.995	0.943
Five	18	18	0.992	1	0.995	0.982
Six	20	20	1	0.962	0.995	0.927
Seven	19	19	0.99	1	0.995	0.944
Eight	10	10	0.989	1	0.995	0.963
Nine	24	24	1	0.991	0.995	0.966

```
Speed: 0.3ms preprocess, 3.6ms inference, 0.0ms loss, 6.8ms postprocess per image
Results saved to runs/detect/train
Ultralytics 8.3.55 Python-3.10.12 torch-2.5.1+cu121 CPU (Intel Xeon 2.20GHz)
Model summary (fused): 168 layers, 3,007,403 parameters, 0 gradients, 8.1 GFLOPs
```

همانگونه که مشاهده میشود، با توجه به حجم کم دیتاهای لیبیل شده و نتیجه نهایی مدل و همچنین پیچیدگی ساختار مدل مشخصا مدل دچار اورفیت شده است که تست ویدئویی نیز این حقیقت را تایید میکند. مشخصا با آموزش مدل با حجم دیتاهای اصلی نتایج به مراتب دقیقتر و بهتری را از آن میتوان انتظار داشت.

در پایین نیز چند مدل از خروجی آمده است، همانطور که ویدئوی پیویت شده میتوانید مشاهده کنید مدل در تشخیص برخی حالات نزدیک بهم، مانند هفت و هشت، در کمال تعجب تا حدی خوب کار میکند اما از تشخیص برخی مقادیر، مشخصا اعداد نه و شش، کاملا باز میماند.





لینک گوگل درایو جهت دسترسی به دیگر اطلاعات پروژه و همچنین کدهای پروژه:

https://drive.google.com/drive/folders/1XNqWvHGzdnZYzOxW•MyAFuFuKDjdH۳fl?usp=drive_link