YOLOv8 re-implementation using PyTorch

Installation

```
conda create -n YOLO python=3.10
conda activate YOLO
conda install pytorch torchvision torchaudio cudatoolkit=11.6 -c
pytorch-lts
pip install opencv-python==4.5.5.64
pip install PyYAML
pip install tqdm
```

Train

- Configure your dataset path in `main.py` for training
- Run `bash main.sh $ --train` for training, `$` is number of GPUs

Test

- Configure your dataset path in `main.py` for testing
- Run `python main.py --test` for testing

Results

Version	Epochs	Box mAP	Download
v8_n	500	37.0	model
v8_n*	500	37.2	model
v8_s*	500	44.6	model
v8_m*	500	50.0	model
v8_l*	500	52.5	model
v8_x*	500	53.5	model

PROF

- `*` means that weights are ported from original repo, see reference
- In the official YOLOv8 code, mask annotation information is used, which leads to higher performance

Dataset structure

```
├── COCO
    ├── images
        ├── train2017
            ├── 1111.jpg
            ├── 2222.jpg
        ├── val2017
            ├── 1111.jpg
            ├── 2222.jpg
    ├── labels
        ├── train2017
            ├── 1111.txt
            ├── 2222.txt
        ├── val2017
            ├── 1111.txt
            ├── 2222.txt
```

Reference

- https://github.com/ultralytics/yolov5
- https://github.com/ultralytics/ultralytics

Things I have changed

File: utils/dataset.py

```
@staticmethod
def collate_fn(batch):
    samples, targets, shapes = zip(*batch)
    for i, item in enumerate(targets):
        # Move existing columns right by 1 and insert batch index at
front
        if len(item):
            item_copy = item.clone()
            item[:, 1:] = item_copy[:, :-1]  # Shift existing columns
right
            item[:, 0] = i  # Add batch index in first column
    return torch.stack(samples, 0), torch.cat(targets, 0), shapes
```