目录

[ГЛАВА 4 Анализ полученных результатов （结果分析） 1](#_Toc134969713)

[ReID模型在其他数据集上的测试结果 1](#_Toc134969714)

[在Market1501、DukeMTMC数据集上的测试结果 1](#_Toc134969715)

[DeepSORT模型的评价指标 2](#_Toc134969716)

[在MOT20数据集上的测试结果 2](#_Toc134969717)

# ГЛАВА 4 Анализ полученных результатов （结果分析）

## ReID模型在其他数据集上的测试结果

### 在Market1501、DukeMTMC数据集上的测试结果

在完成了模型的训练之后，我们又选择在另一个数据集Market1501、DukeMTMC上测试我们的模型性能，并与同时期出现的其他reid模型做了横向对比：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Methods | Market1501 | | DukeMTMC | |
| Rank-1 | mAP | Rank-1 | mAP |
| IANet(IVPR’19) | 94.4 | 83.1 | 87.1 | 73.4 |
| Auto-ReID(ICCV’19) | 94.5 | 85.1 | - | - |
| OSNet(ICCV’19) | 94.8 | 84.9 | 88.6 | 73.5 |
| ABDNet(ICCV’19) | 95.6 | 88.3 | 89.0 | 78.6 |
| Circle Loss(CVPR’20) | 96.1 | 87.4 | 89.0 | 79.6 |
| **ours** | **95.7** | **88.4** | **90.1** | **81.3** |

根据上面提供的数据，我们可以看到FastReID模型在Market1501和DukeMTMC两个数据集上都表现出色。其在Market1501数据集上的Rank-1精度为95.7%，mAP为88.4%，在DukeMTMC数据集上的Rank-1精度为90.1%，mAP为81.3%。

相比于其他算法，FastReID取得了很好的性能表现。例如，在Market1501数据集上，FastReID的Rank-1精度和mAP分别超过了IANet、Auto-ReID、OSNet等算法；甚至在DukeMTMC数据集上，它的Rank-1精度和mAP超过了其他所有的算法。这说明FastReID在行人重识别任务上具有很强的鲁棒性和泛化能力。

总之，基于这些评估指标，我们可以认为FastReID是一种有效的行人重识别模型，有望应用于实际场景中。

## DeepSORT模型的评价指标

### 在MOT20数据集上的测试结果

为了验证基于fastreid改进的deepsort多目标跟踪方法的有效性，我们在又在更具挑战性的新数据集MOT20上展开了对比实验。

该数据集使用静态和移动相机拍摄的不受约束的环境中注释了 8 个具有挑战性的视频序列（4 个训练，4 个测试）。相比于此前的MOT Challenge系列数据集，MOT20关注人群密集的场景，其视频最多可达单帧 246 人。所以跟踪显得更加困难。它比较考验多目标跟踪模型对小目标的追踪能力。

许多人在商店里

中度可信度描述已自动生成

图表 1 MOT20-01截取片段

我们将多目标跟踪算法放到在MOT20数据集中四个子集上进行了测试，并得到的结果与世界范围的MOT Challenge 比赛中的前10名优胜者作比较，结果如下表：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Traker | **MOTA ↑** | **IDF1↑** | **MT/% ↑** | **ML/% ↓** | **IDs ↓** | **FPS ↑** |
| ByteTrack | 67 | 70.2 | 47.7 | 21.2 | 680 | 17.7 |
| OUTrack | 65.4 | 65.1 | 49.5 | 13.3 | 2,885 | 5.1 |
| UTM | 64.4 | 65.9 | 65 | 8.7 | 2,592 | 22.4 |
| RFTracker | 62.4 | 53 | 48.7 | 15.5 | 3,804 | 0 |
| SUSHI | 61.6 | 71.6 | 47.6 | 19.2 | 1,053 | 5.3 |
| TransCenter | 61 | 49.8 | 48.4 | 15.5 | 4,493 | 1 |
| MPTC | 60.6 | 59.7 | 51.1 | 16.7 | 4,533 | 0.7 |
| TMOH | 60.1 | 61.2 | 46.7 | 17.8 | 2,342 | 0.6 |
| OCSORT | 59.9 | 67 | 38.5 | 26.6 | 554 | 27.6 |
| MFI | 59.3 | 59.1 | 41.1 | 17.3 | 191 | 0.5 |
| **ours** | **60.4** | **60.9** | **35.2** | **17.4** | **2021** | **8.6** |

首先，需要说明的是在MOT20数据集上，跟踪器的表现被衡量为MOTA、IDF1、MT和ML等指标。其中，MOTA是多目标跟踪的主要评估指标，它综合了检测、匹配和跟踪的精度和召回率。IDF1是一种更加注重跟踪器的准确性的指标，MT和ML分别代表正确跟踪的目标数量占总数的比例和漏跟踪目标数量占总数的比例，IDs表示错误标识符数量，FPS表示每秒能够处理的帧数。

从数据可以看出，ByteTrack表现最好，其次是OUTrack和UTM，这三个跟踪器的MOTA均超过了64。其他跟踪器的表现相对较差，但也有一些跟踪器在某些指标上表现突出。例如，SUSHI和OCSORT在IDF1和MOTA上得分较高，但是在ML方面表现较差，而TMOH则在IDF1上表现较好。

我们的跟踪器对比其他10名世界范围内的优胜者，有如下的优点和缺点：

优点：

* 该跟踪器在IDF1指标上表现不错，得分60.4，说明其基本能够完成多目标跟踪任务。
* 该跟踪器在FPS方面表现也比较良好，达到了8.6帧/s，这意味着它可以处理高速动态场景。

缺点：

* 在MOTA指标上，该跟踪器只有60.4的得分，说明它在数据集MOT20的表现上相对较弱。
* ML/ %指标较高，为17.4，说明它对于漏跟踪目标数量占总数的比例也没有很好的控制。
* 该跟踪器在IDs两个指标上的表现都不如其他一些跟踪器，IDs为2021，MT/ %为35.2，这说明它存在一定的误检、漏检和误跟踪等问题。

虽然我们的跟踪器在FPS和IDF1指标上表现不错，并且具有一定的实时性，但在正确跟踪目标数量占总数的比例和漏跟踪目标数量占总数的比例上表现相对较差。经过分析，我认为表现差和YOLO目标检测算法有关。我们的YOLO检测算法属于一阶段的目标检测算法，优点是运行速度很快，节省计算资源。而缺点也非常明显：位置精确性差,对于小目标物体以及物体比较密集的也检测不好,比如一群小鸟，一大群密集人群。而在前10名中，除了OCSORT、ByteTrack和UTM 之外，都是用了二阶段的目标检测算法，这保证了他们针对小目标也有较高的目标检测质量。

因此我认为，如果后续要提升模型对小目标的追踪能力，可以从YOLO入手，有以下策略：

* 调整模型结构：可以考虑使用更深的网络结构，以提高模型的识别能力。
* 多尺度检测：可以设计多尺度检测策略，对不同大小的目标进行识别和检测，提高模型的检测精度。例如在YOLO网络的颈部（neck）增加FPN（Feature Pyramid Networks）结构，将顶层特征和底层特征融合，以提高网 络检测不同尺度目标的能力。