

## 3.5 图像分类数据集重点摘录与练习解答

### 问题解答

#### 1、减少batch\_size（如减少到1）是否会影响读取性能？

解：较大的 batch\_size 可以利用并行化操作来提高数据加载的效率。在读取数据时，可以同时加载多个样本到内存中，以减少磁盘读取数据的时间。而当 batch\_size 减少到1时，无法利用并行化操作，每次只能加载一个样本，从而导致数据加载的效率降低。

此外，较小的 batch\_size 也会带来更多的数据加载次数。数据加载在计算机中属于 I/O 操作，通常是相对较慢的。因此，较小的 batch\_size 会导致更频繁的数据加载操作，增加了整体的读取时间。

然而，较小的 batch\_size 也具有一些优点，比如更快的模型更新和更精确的梯度估计。因此，在选择 batch\_size 时，需要综合考虑性能和模型训练的需求，权衡不同因素。

#### 2、数据迭代器的性能非常重要。当前的实现足够快吗？探索各种选择来改进它

解：

- 使用 pin\_memory=True：在创建数据加载器时，设置 pin\_memory=True 来将数据加载到 GPU 内存中，利用GPU的并行性来加快数据传输速度。
- 使用多个 get\_dataloader\_workers：通过增加 get\_dataloader\_workers 参数的值，可以使用多个子进程或线程来并行读取数据，从而加快数据加载的速度。可以根据计算机的硬件配置和数据加载的需求，选择合适的 get\_dataloader\_workers 值。

#### 5、查阅框架的在线API文档。还有哪些其他数据集可用？

解：

```
dir(torchvision.datasets)
```