

光神经网络实验报告

陈原、陈敬徐

2025 年 5 月 18 日

一、实验原理

光神经网络全称为全光衍射深度神经网络，通过光的传播与干涉实现数据的传输与线性求和。在光神经网络中，可以通过调节中间层对不同位置接收到的光信号进行相位调控，进而调节光的相互干涉。因此，我们可以将希望实现的权重以相位的形式传入光神经网络，从而达到权重的调节，实现神经网络。

在光神经网络的搭建中，用到了两个重要的光学元件。数字微镜器件 DMD 是一个反射镜阵列，由许多能够偏转 0° 、 12° 、 -12° 度的反射镜构成。通过控制 DMD 每个反射镜的偏转状态，我们可以调节反射到特定角度的光的图案，作为输入层。空间光调制器 SLM 是一个液晶单元阵列，每个液晶单元都能加上不同的电极电压以调节折射率，改变光程，实现调控特定位置光的相位。将特定的相位阵列输入给 SLM，就可以调节权重，因此 SLM 是光神经网络中的中间层。

二、实验过程记录

我们搭建了光神经网络系统，成功在 CCD 上观察到聚焦后的像。我们主要参考黄俊涵调节光路的方法，主要步骤不再赘述。这里就我们解决的几个问题做说明。

在初调光路时，我们解决了 DMD 光路难以调节的问题。由于 DMD 在所有反射镜平行时相当于闪耀角为 12° 的闪耀光栅。我们观察到无论如何调节 DMD 入射光的角度均无法使得出射光朝向 SLM 方向。我们发现，这是由于闪耀光栅的法线方向与垂直 DMD 方向有 12° 的夹角，入射、出射方向始终位于法线方向

的两端。如图 1 所示，当闪耀光栅法线方向为 n_2 时，无论如何调节均无法得到正确光路。最后，我们调节闪耀光栅朝向，使 DMD 全都偏向另一个方向，完成了光路的调节。

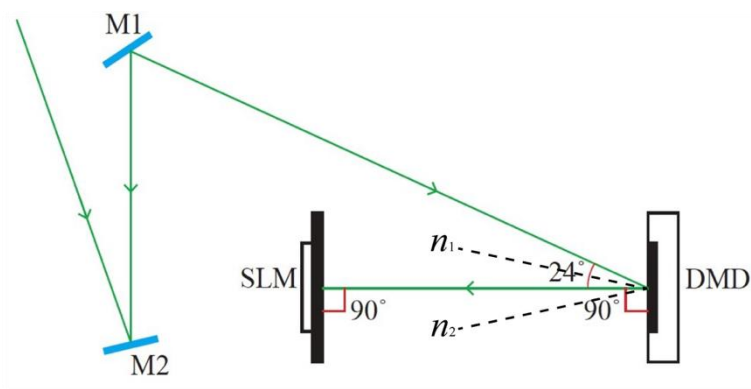


图 1. 闪耀光栅不同朝向的示意图

此外，在调节扩束镜的过程中，为了避免其俯仰角导致光路的偏折，我们更换了扩束镜的底座，使其能够在一定范围内调节俯仰角度，这使得我们的光路更加水平。

三、实验结果的分析与讨论

1. 方形图片以及双缝干涉的成像观察

当 DMD 上为方形图案时，我们在 CCD 上观察到了其像。当 DMD 上为双缝时，我们在 CCD 上观察到了其干涉。尽管其还是不够垂直，说明部分光学元件没有完全水平；但整体能够实现双缝干涉，说明光路调节的效果较好。

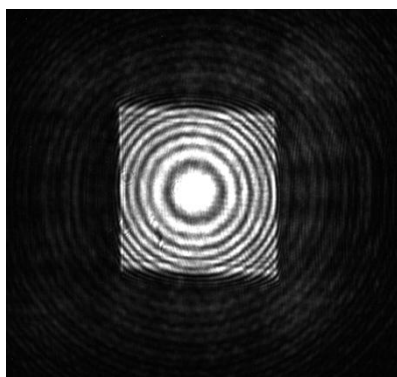


图 2. 方形图案的像

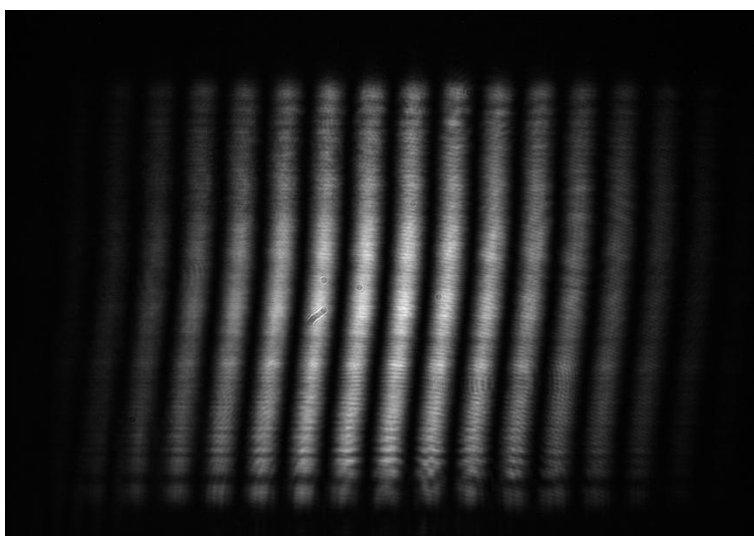


图 2. 双缝图案的干涉像

2. Gamma 矫正

我们对图像进行 Gamma 矫正。我们自动扫描了灰度为 0-255 范围内的干涉图像，选择在第 220 行第 170-180 列的数据点进行傅里叶变换。得到的 Gamma 矫正曲线如图 3 所示。我们观察到在 π 和 $-\pi$ 之间的近线性分布以及 2π 的跃变。这说明此时仪器的灰度符合线性。我们选择第 170-180 列的数据点的原因是，在这个区间内，峰出现了明显的偏移，且各峰幅值接近，如图 4 所示。

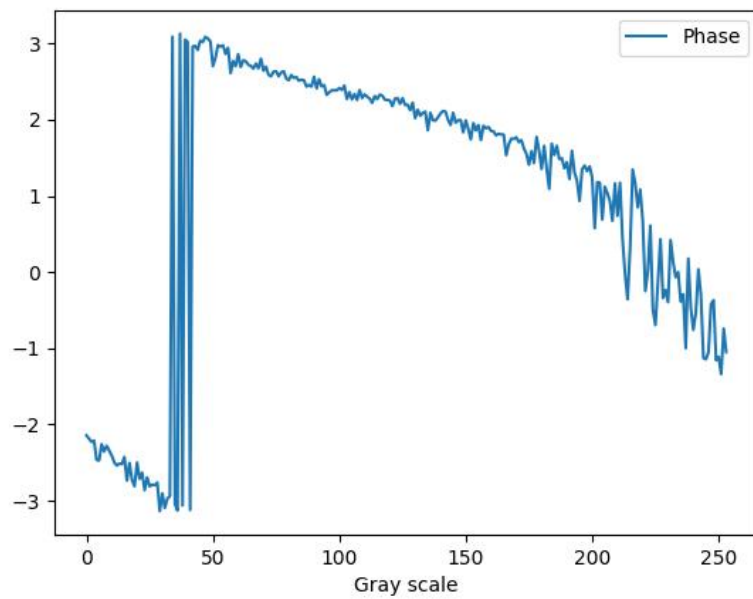


图 3. Gamma 矫正曲线

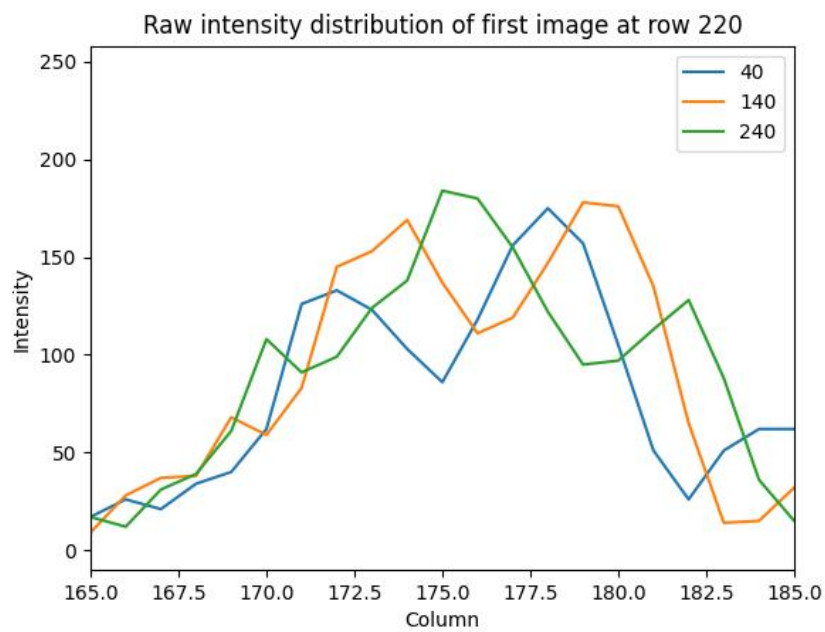


图 4. 第 170-180 列不同灰度值的峰的分布（右上角为特定曲线的灰度值）

这里我们还观察到，若选取的列数过长，相位曲线不再连续。我们猜测这是由于选取范围内峰的不同幅值影响了傅里叶变换过程。由于包络线并非水平线，不同峰由于其幅值不同，在傅里叶变换中的权重也不同，影响了相位的计算。

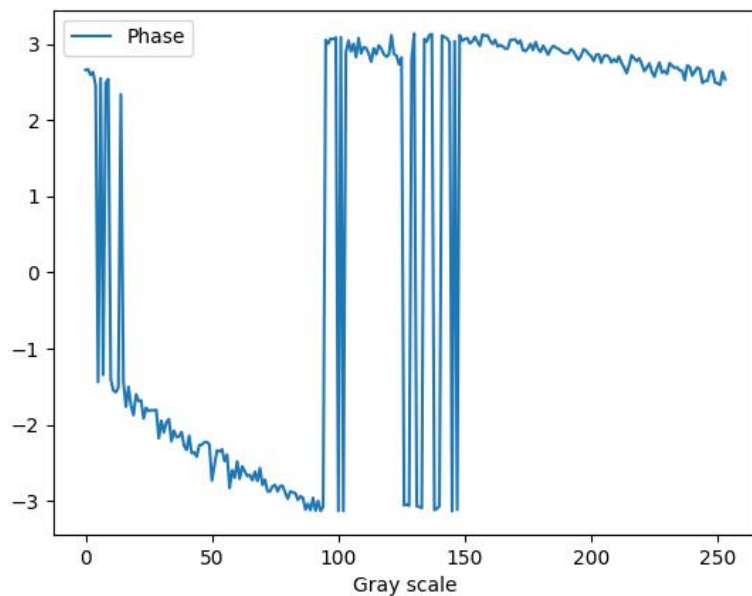


图 5. 数据列为 30 列的 Gamma 矫正曲线

3. 参数的测量

在焦距测量部分，我们得到图像最锐利时的焦距为 0.2424 m；我们还用缝宽 6 个像素、双缝间距 120 个像素拍摄了双缝干涉，通过傅里叶拟合得到如图 6 所示的，间距为 1.071m。因此，我们计算出物距和像距为 0.700 m 和 0.371 m。考虑到装置的几何构型，我们认为物距是 0.700 m 而像距是 0.371 m。需要说明的是，在完成对焦后为了观察双缝干涉图样的不同位置，我们移动了 CCD，这可能导致实际的物距、像距出现偏差。

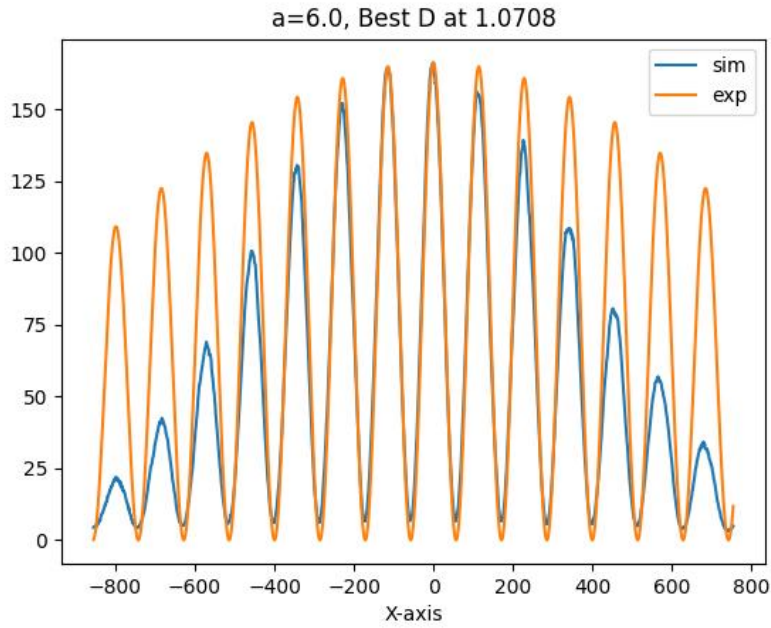


图 6. 双缝干涉的拟合曲线

四、DNN 神经网络训练模拟

DNNModel.py 中定义了具体的传输模型，D2NN 采用连续相位调制，基于角谱法在空间域和频域间模拟光场传播，同时支持振幅和相位调制；FD2NN 全程在频域计算，减少傅里叶变换次数并支持外部相位注入，计算效率更高，适合大规模数据处理或实时光学系统，如高速图像识别或频域光学信号处理；D2NNBinary 将相位参数二值化，简化调制器设计，继承自 D2NN 但固定为相位调制，适用于对硬件实现成本敏感的场景，如低功耗光学传感器或集成光学芯片，但可能因相位离散化导致精度损失。后续引用的模型为 D2NN。

用 TrainD2NN.py 训练。d1、d2 值为之前测得的衍射参数 u 、 v ；输入图片的像素大小 dimension，训练相位的像素大小 Phase_dimension，MSE 以及 SCE 损失函数的占比等参数不变。训练结束得到一个相位矩阵，保存。

用 ShowD2NN.py 可视化训练结果，如图 7。回到实验中，再将训练相位 phase.npy 传输到 SLM 上，即可实现光神经网络。

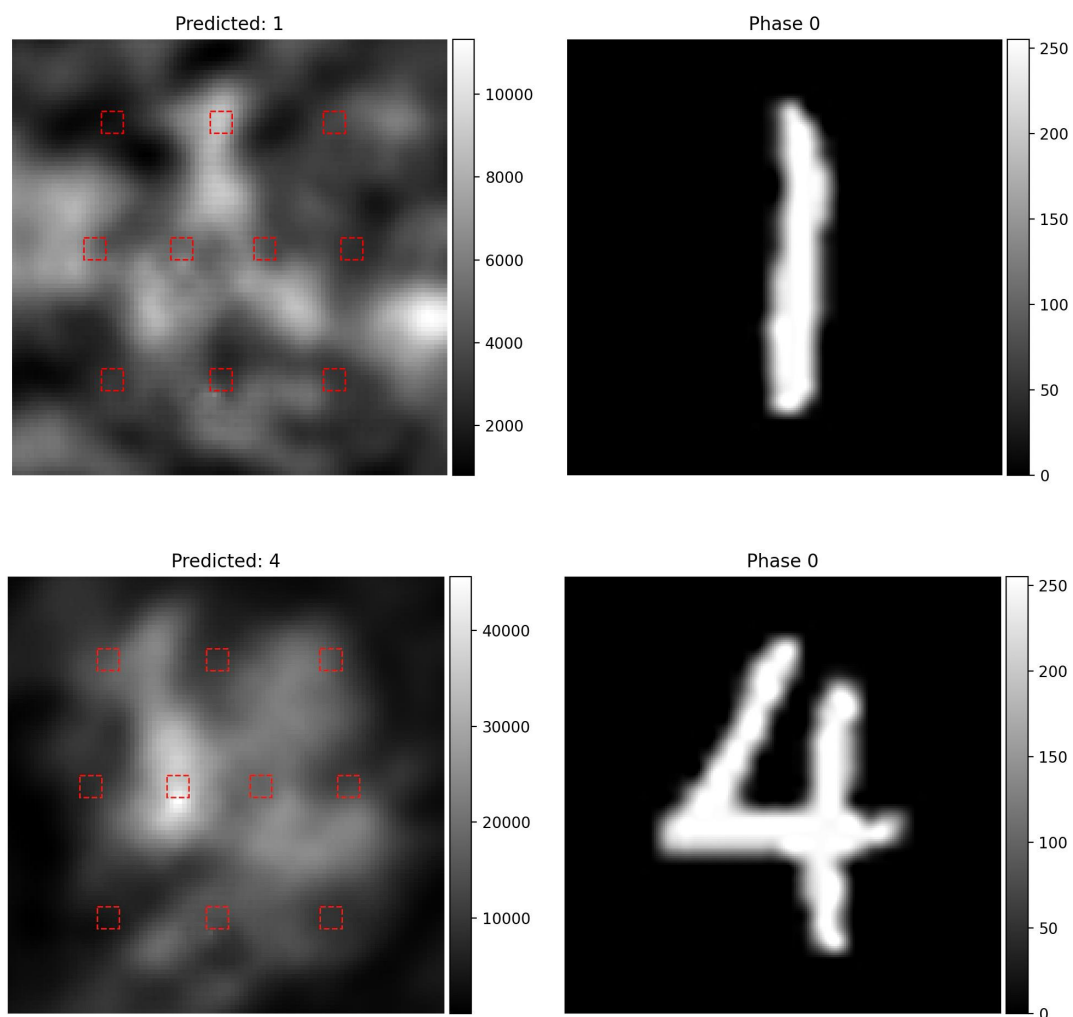


图 7. 训练结果可视化示例