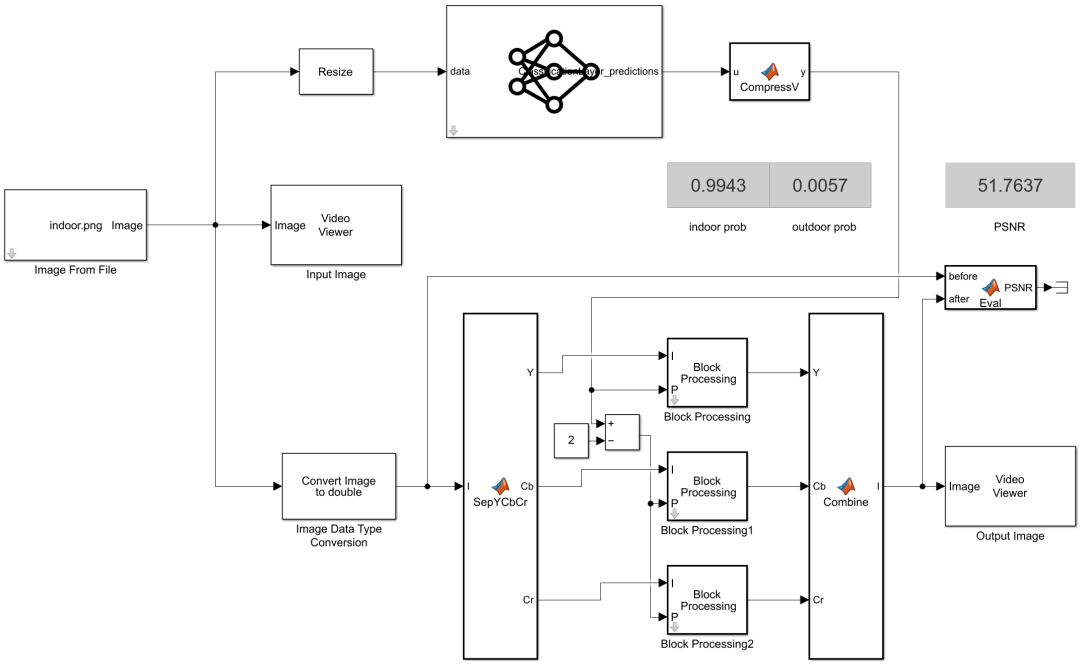
实现报告



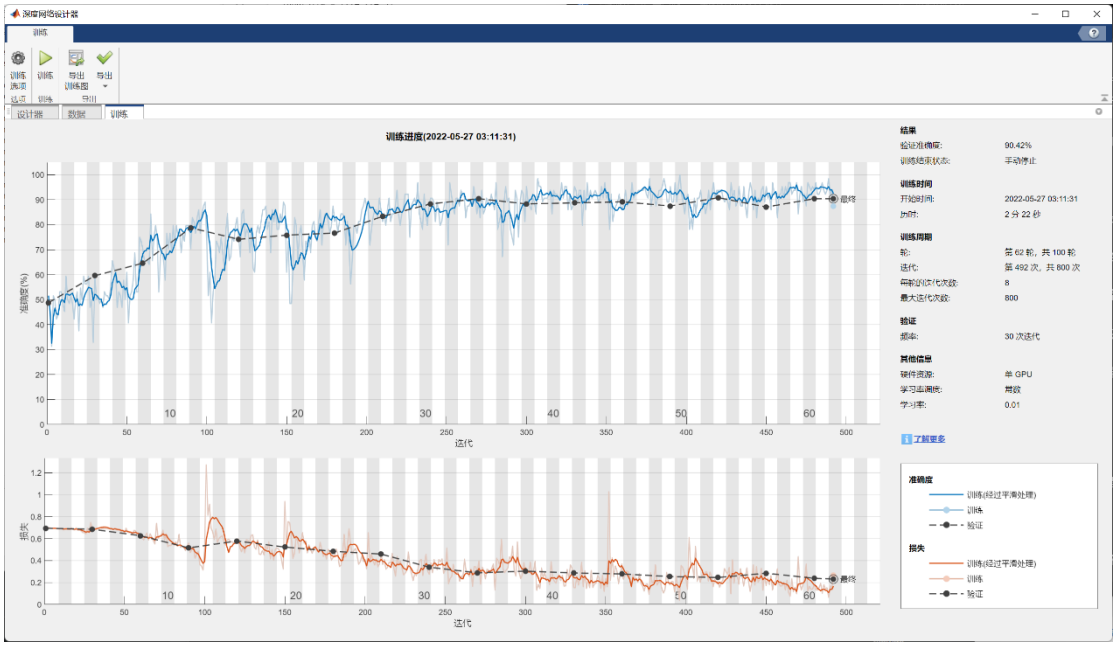
使用到的组件：

- MATLAB R2022a
- Simulink
- DSP System Toolbox
- Image Processing Toolbox
- Computer Vision Toolbox
- Deep Learning Toolbox
- Deep Learning Toolbox Model for ResNet-50 Network(仅在训练时需要)
- Parallel Computing Toolbox, GPU Coder(可选，为了 GPU 加速训练)

在 MATLAB 中：训练神经网络分类器

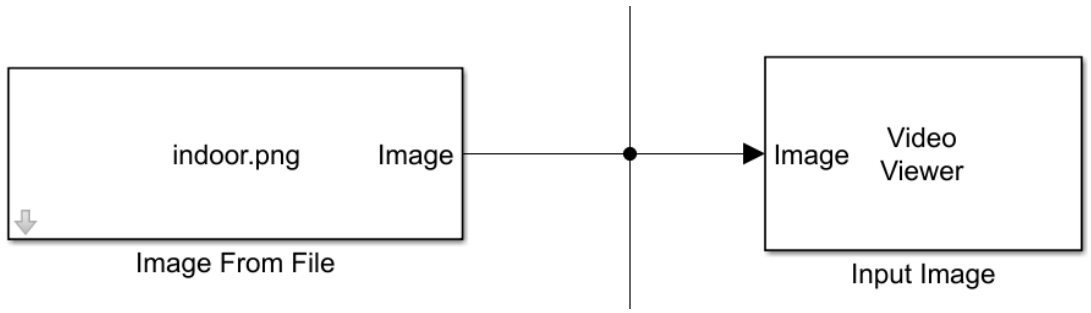
实现程序位于 MATLAB_Classifier_Training\Classifier.m

- 从该网站下载图像训练集，解压于 images 文件夹。训练集有两个子文件夹，indoor 代表室内照片，outdoor 代表室外照片，每个类别均有 400 张。将每张照片拉伸并缩放到 224x224 分辨率，存于 images_resized224 文件夹，保留目录结构。
- 使用 MATLAB 中的 imageDatastore()函数读入训练集，该函数可根据子文件夹名添加图像标签。
- 使用 MATLAB 的“深度网络设计器”功能导入 ResNet-50 神经网络，并将代码导出到 getLayers.m 函数中。修改第一层输入层和倒数第二层全连接层的大小分别为 224x224x3 和 2，以匹配输入图像分辨率和输出的两种类型(室内和室外)。
- 尽管可以直接使用 MATLAB 内的 trainNetwork()函数来训练网络并保存，但为了能方便地在 Simulink 中导入模型，我直接使用“深度网络设计器”功能来训练网络，如



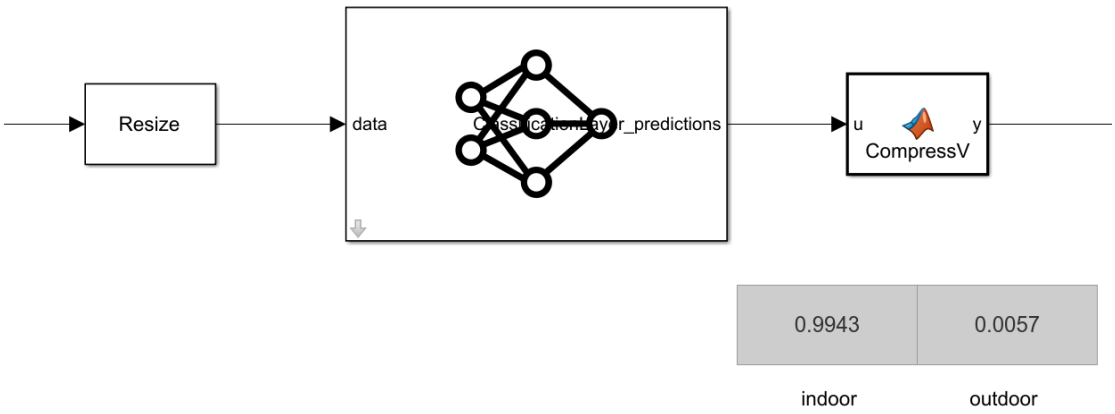
图所示：之后选择将模型导出到 Simulink，即可获得可在 Simulink 中使用的图像分类模型：trainedNetwork_224x224.mat。

读取图像



使用 Image From File 模块读入图片，并立即显示以方便后续图像对比。

分类图像并决定压缩程度

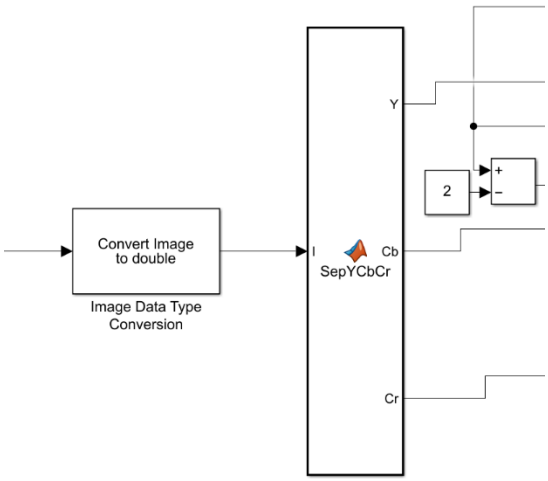


由于训练的深度神经网络的输入大小为 224x224x3，因此先将输入的图像缩放到 224x224 分辨率，再传入神经网络予以分类。神经网络输出一个 1x2 的向量，第一个值表示该图像为室内照片的概率，第二个值表示该图像为室外照片的概率，两个值的和为 1。在上图的示例中，输入图像为我的寝室内的照片，因此认为照片为室内照片的概率为 99.43%，可认为已被准确分类。

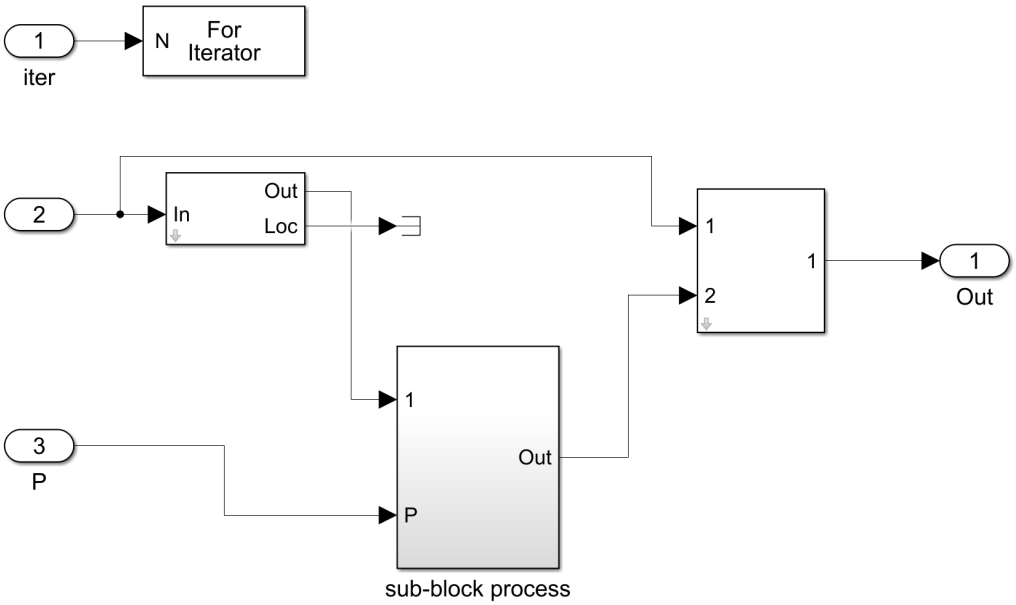
之后将分类结果传入 MATLAB 函数 CompressV，该函数在图像分类为室内时输出压缩系数 12，分类为室外时输出 6，并传输到之后的 DCT 后压缩使用的掩码生成器中，以生成对应大小的掩码。值得注意的是压缩系数在传入到 Cb、Cr 两个色度分量时被减去了 2，这是因为人眼对明度敏感性不如色度，因此适当减少色度信息有助于更高效地压缩。

提取图像 YCbCr 通道

先将图像由 int8 类型转换到 double 以便后续处理，之后使用 MATLAB 函数 SepYCbCr 将 RGB 格式转换成 YCbCr 格式，并分通道输出。

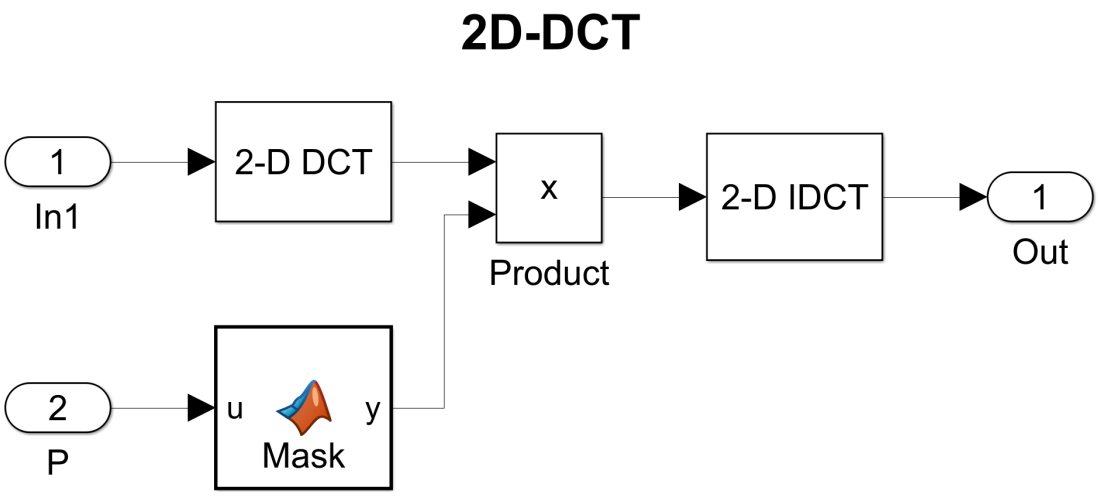


图像区块分割与合并



为了对整个图像进行分块 DCT,该实验使用了 Computer Vision Toolbox 中的 vipmisc 库(MATLAB\R2022a\toolbox\vision\vision\vipmisc.slx)里的分块与拼接。该库提供了 Input 和 Output 一对模块，可以用自定义的区块大小(本实验中为 8x8)分割图像并在处理后拼接。在上图所示子系统中，为了遍历整个图像，该子系统运行了区块个数次。

区块的 2D-DCT 压缩



在该子系统中，输入的是每个 8x8 的区块。在 In1 输入区块图像后，对其应用 DCT，使得图像的高频信息靠近右下角，低频信息靠近左上角。

端口 P 传入了由分类器和通道类型决定的压缩系数，根据该系数使用 MATLAB 函数 Mask 生成掩码。根据压缩系数，决定了有多少个右上-左下对角行填 1，而其余的填 0，如右图所示；将该掩码与应用 DCT 后的区块相乘，使得区块中只保留了靠近左上填 1 的低频部分，靠近右下填 0 的高频部分则被舍弃。据此，实现了根据压缩系数决定了保留多少图像高频信息。

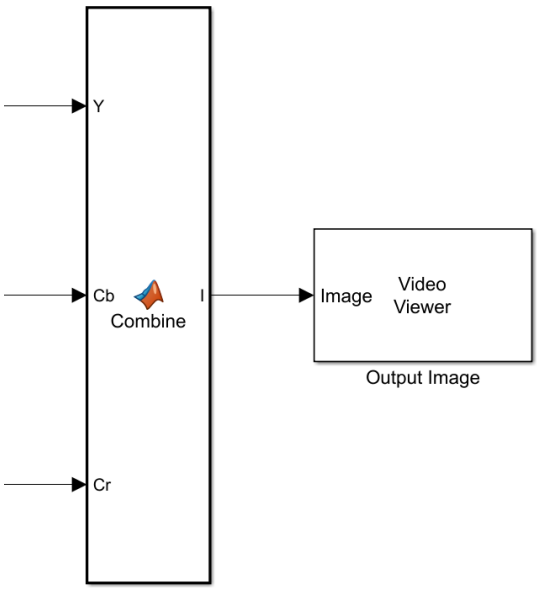
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	0
1	1	1	1	1	1	0	0
1	1	1	1	1	0	0	0
1	1	1	1	0	0	0	0
1	1	1	0	0	0	0	0

压缩系数=10 对应的掩码

对处理后的图像应用 DCT 逆变换，得到了压缩高频信息后的 8x8 区块，并如上述拼接为完整图像。

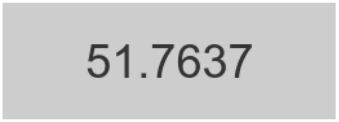
合并图像通道并输出

经过上述处理后得到的是处理后的 YCbCr 通道，因此 MATLAB 函数 Combine 将图像的 Y、Cb、Cr 三个通道进行了合并，并转化成 RGB 格式后输出图像。该图像即为经过 DCT 与压缩处理后的完整图像，本实验流程到此结束。



计算损失

在 MATLAB 函数 Eval 中使用 psnr()方法来计算处理后的图像与原图相比的峰值信噪比(Peak signal-to-noise ratio)。使用 Display 模块监听 PSNR 输出，该值越大，代表输出图像与原图像相比损失较少；该值越小，代表输出图像与原图像相比损失较大。



PSNR