

化工与人工智能研究生课程，2023秋季

化工与人工智能

Chemical Engineering and Artificial Intelligence

霍 锋
刘亚伟

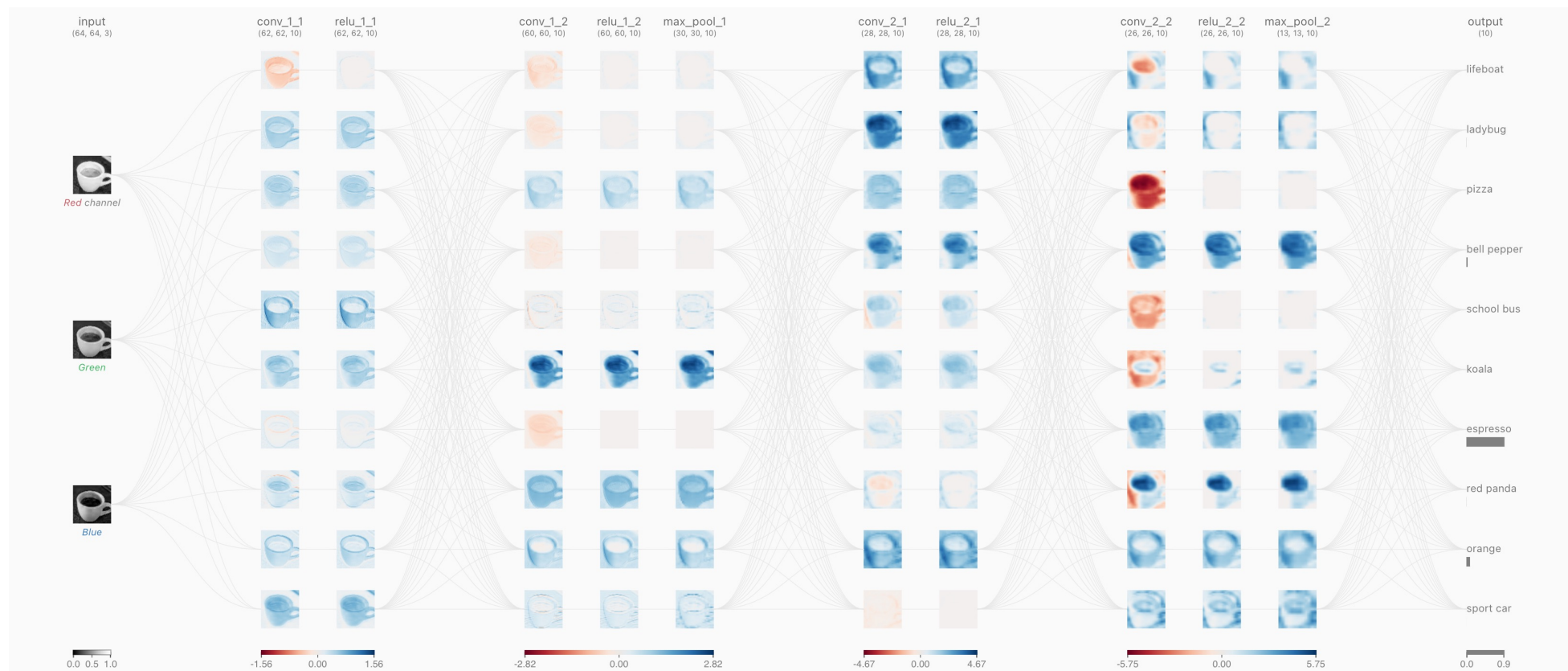
中国科学院过程工程研究所

第六章 深度学习

卷积神经网络, Convolutional Neural Network, CNN

1.1 什么是CNN：概念

一个用于图片分类的CNN： [CNN Explainer \(poloclub.github.io\)](https://poloclub.github.io/cnn-explainer/)



通过**卷积层**、**池化层**自动提取特征、并通过多层结构进行信息处理的深度学习模型，广泛应用于图像和视频分析。

1.2 什么是CNN：历史发展

➤ 早期发展（1980s）：

- ✓ 1980年：由日本学者Fukushima提出的Neocognitron是第一个提出类卷积神经网络的模型，具有多层感知机结构，用于手写字符识别。
- ✓ 1986年：LeCun等人提出了更为现代化的卷积神经网络架构，并使用反向传播算法来训练网络，这使得神经网络能够自动调整权重以减少误差，极大地提高了神经网络的表现。

➤ 1990年代应用与发展：

- ✓ 1998年：Yann LeCun等人提出了LeNet-5网络，这是一个具有卷积层和池化层的卷积神经网络，成功应用于手写数字识别任务（MNIST数据集）。LeNet-5是第一个广泛应用并取得成功的CNN模型，成为后续研究的基础。

➤ 2000年代研究停滞与小规模应用：

- ✓ 尽管CNN在手写识别等任务上表现优秀，但由于计算资源的限制，CNN的应用未能迅速扩展。此时期，神经网络的研究和应用相对较少，传统机器学习方法（如支持向量机）更为流行。

1.2 什么是CNN：历史发展

➤ 2012年突破性进展（深度学习的复兴）：

- ✓ 2012年：AlexNet的提出标志着CNN的复兴。由Alex Krizhevsky、Ilya Sutskever和Geoffrey Hinton设计的AlexNet在ImageNet竞赛中取得了压倒性的胜利，成功地证明了深度卷积神经网络在大规模图像分类任务中的巨大潜力。该模型深度达8层，采用了ReLU激活函数、dropout正则化以及GPU加速训练等技术，大大提高了训练效率。

➤ 2010s深度CNN的快速发展与应用：

- ✓ 2010年代，深度学习技术和CNN迅速发展并广泛应用于各种领域，包括物体检测、语音识别、自然语言处理等。VGGNet、GoogLeNet、ResNet等新型网络架构的提出，不断推动CNN的性能提升。

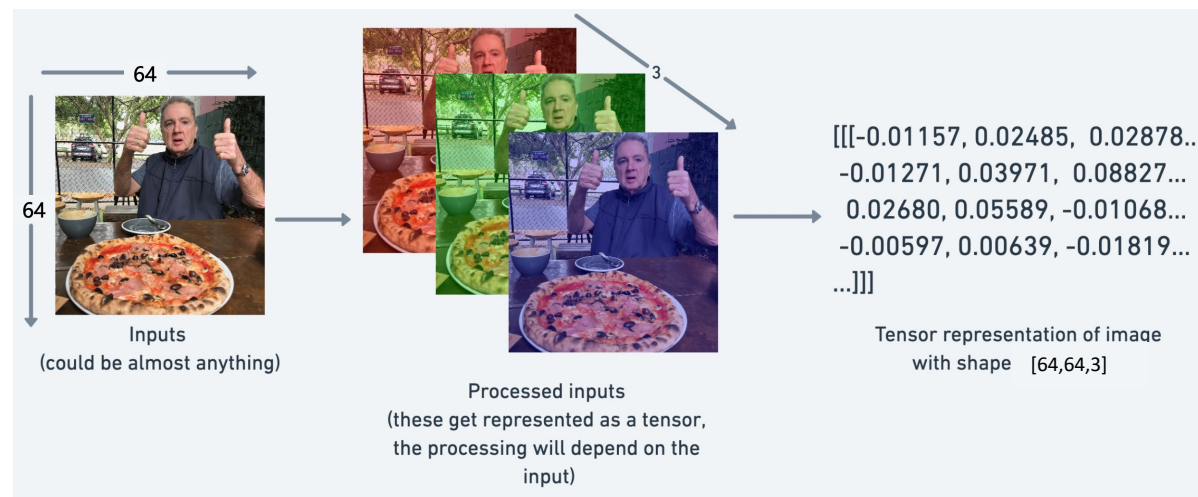
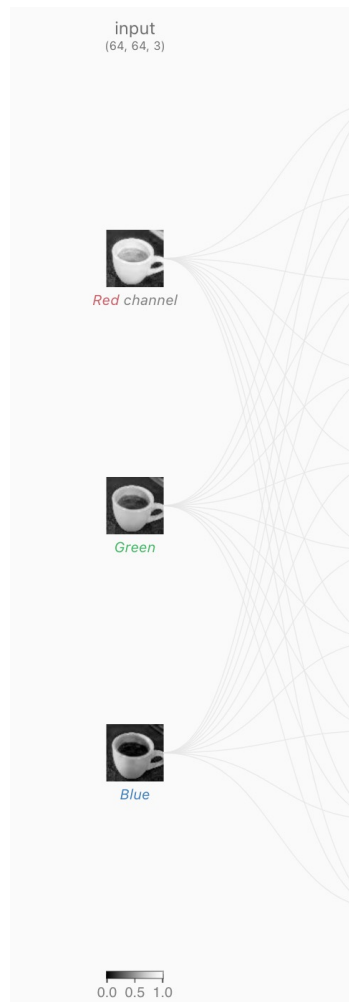
➤ 当今发展

- ✓ CNN如今已经成为**计算机视觉、医学图像分析、自动驾驶、语音识别**等领域的核心技术，并不断推动人工智能的前沿发展。

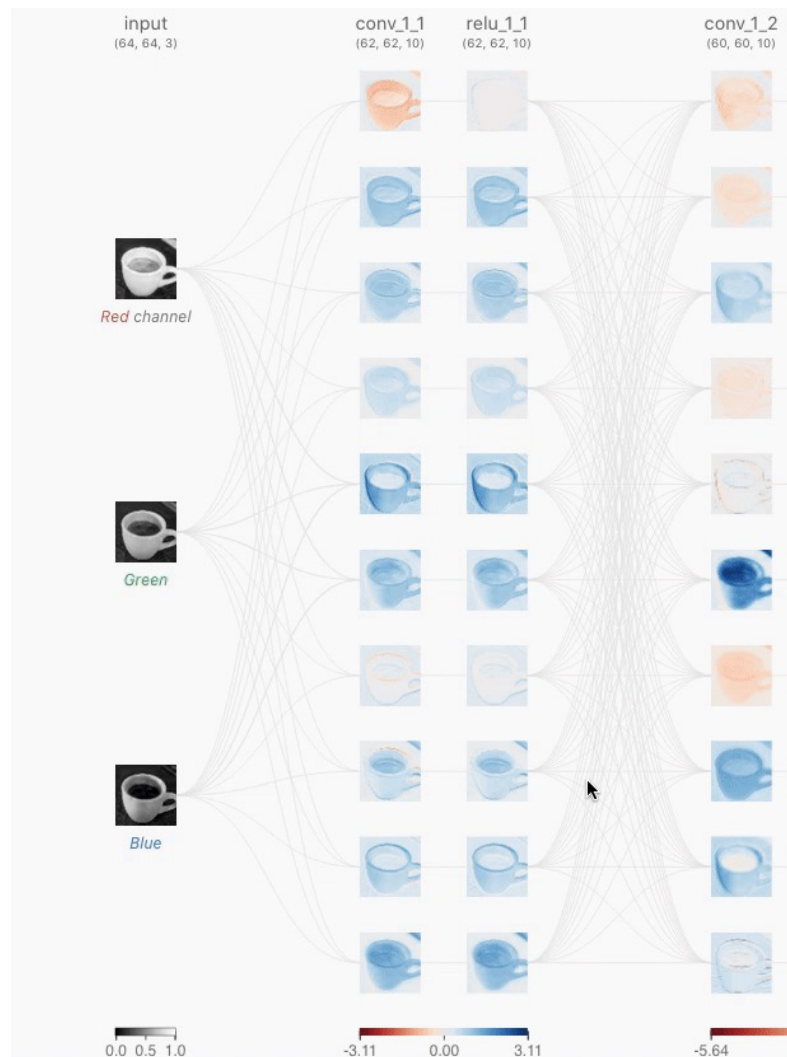
CNN在材料和化工方向的应用主要通过自动化分析材料微观结构、预测物性参数和优化化工过程，助力新材料设计和过程优化。

2.1 CNN的基本结构：输入层

- 输入层（最左边的层）表示 CNN 的输入图像。因为这里使用 RGB 图像作为输入，所以输入层有三个通道，分别对应红色、绿色和蓝色通道，显示在该层中。
- 输入量为一个三维/三阶张量，形状为 $[64*64*3]$

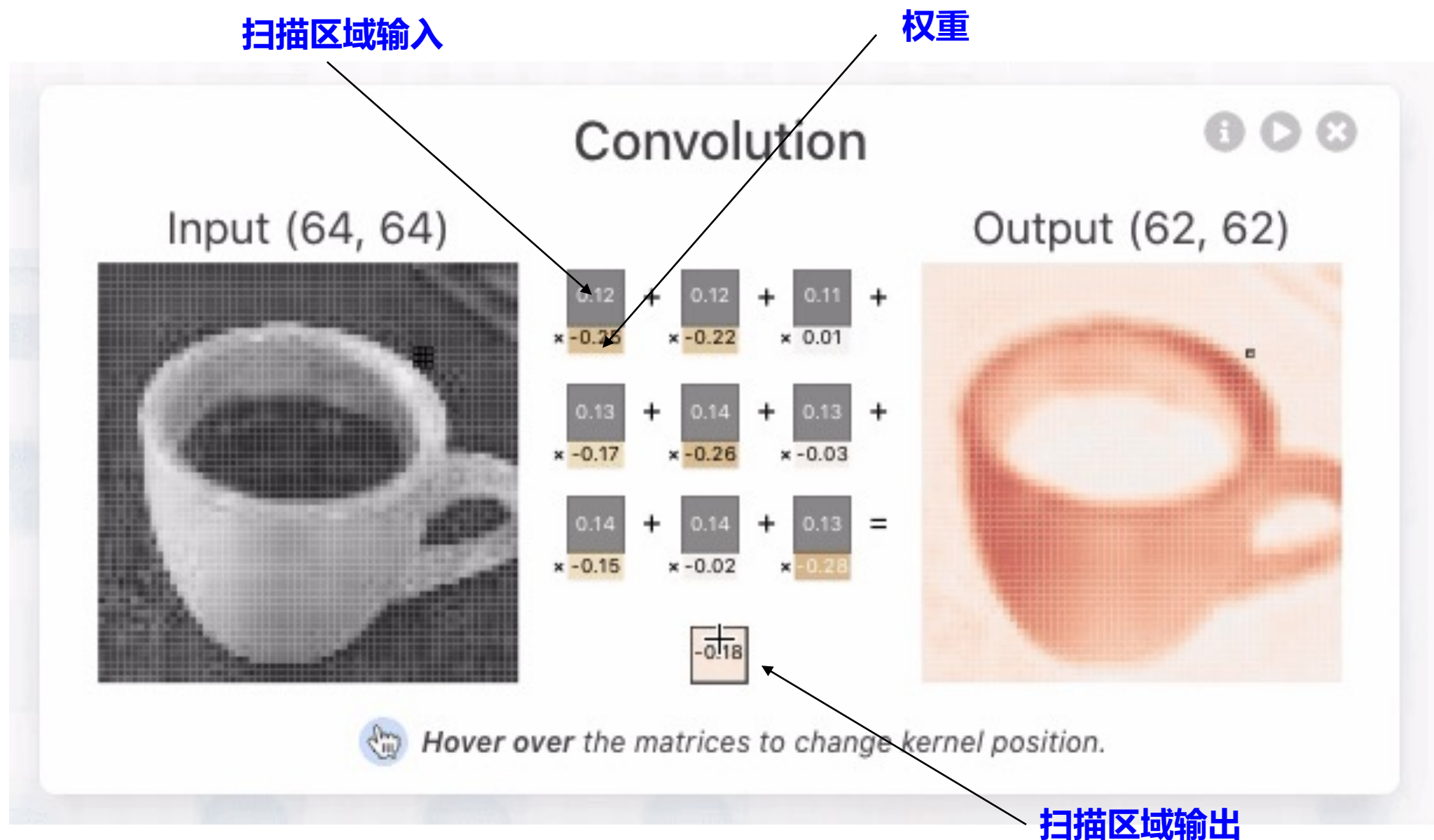


2.2 CNN的基本结构：卷积层

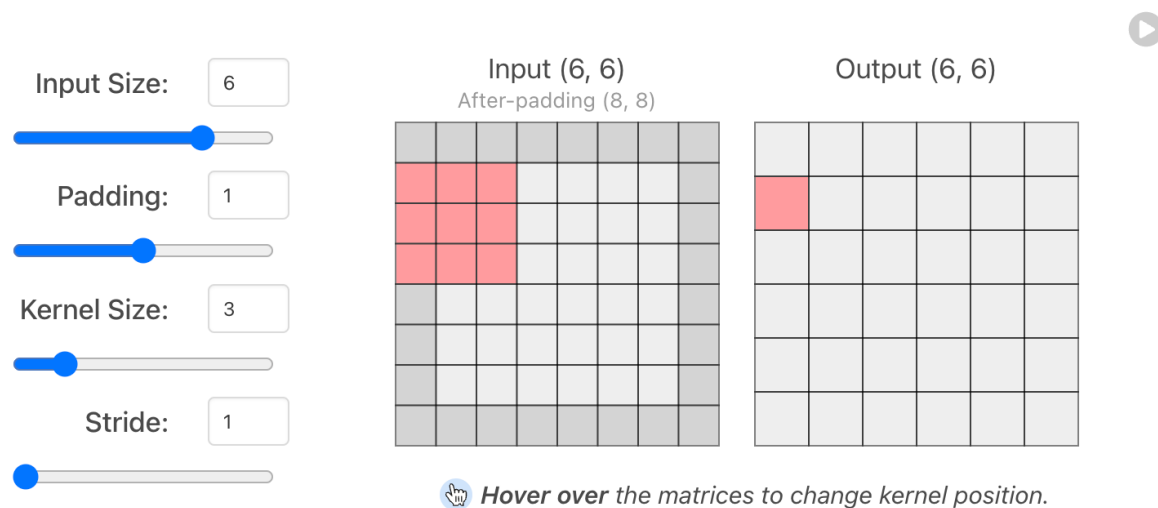


- 卷积层包含需要训练的**卷积核**（类似之前讲的**权重 w** ）。
- 卷积层首先通过卷积核（也叫滤波器）对输入数据进行滑动扫描。卷积核是一个小的矩阵，它在输入图像上逐步滑动，对每个位置进行计算，并生成一个输出特征图（或称为激活图）。卷积操作可以理解为通过加权求和的方式对局部区域进行特征提取。
- 然后，执行元素求和，左图包含所有 3 个中间结果以及网络学习的偏置（类似之前讲的**偏置 b** ）。在此之后，得到的二维张量将是第一个卷积层中神经元的激活图。

2.2 CNN的基本结构：卷积核



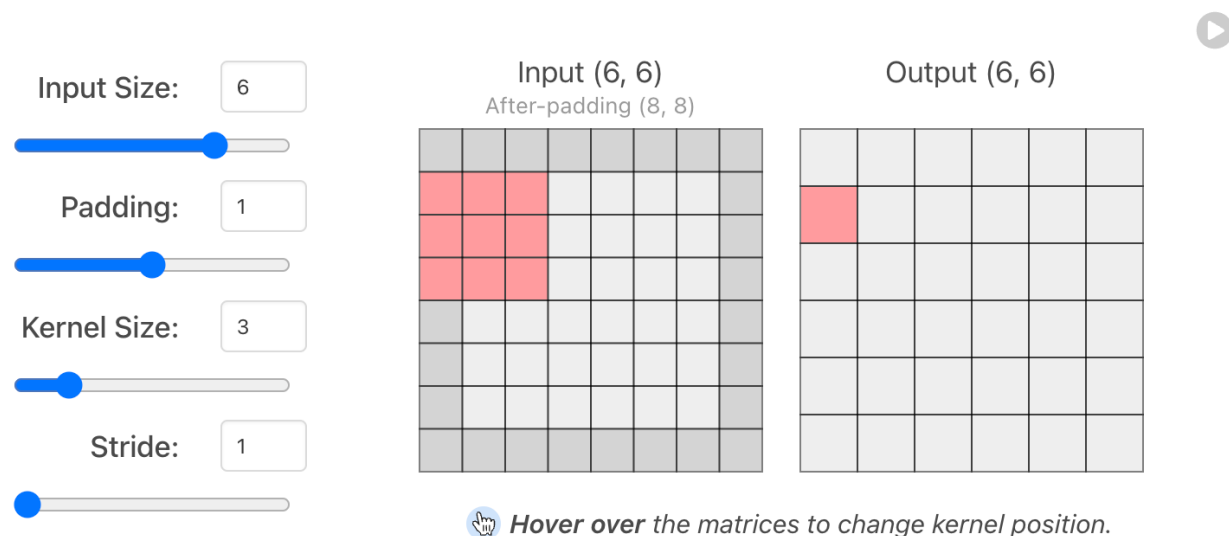
2.2 CNN的基本结构：卷积核



红色区域卷积核大小：3*3

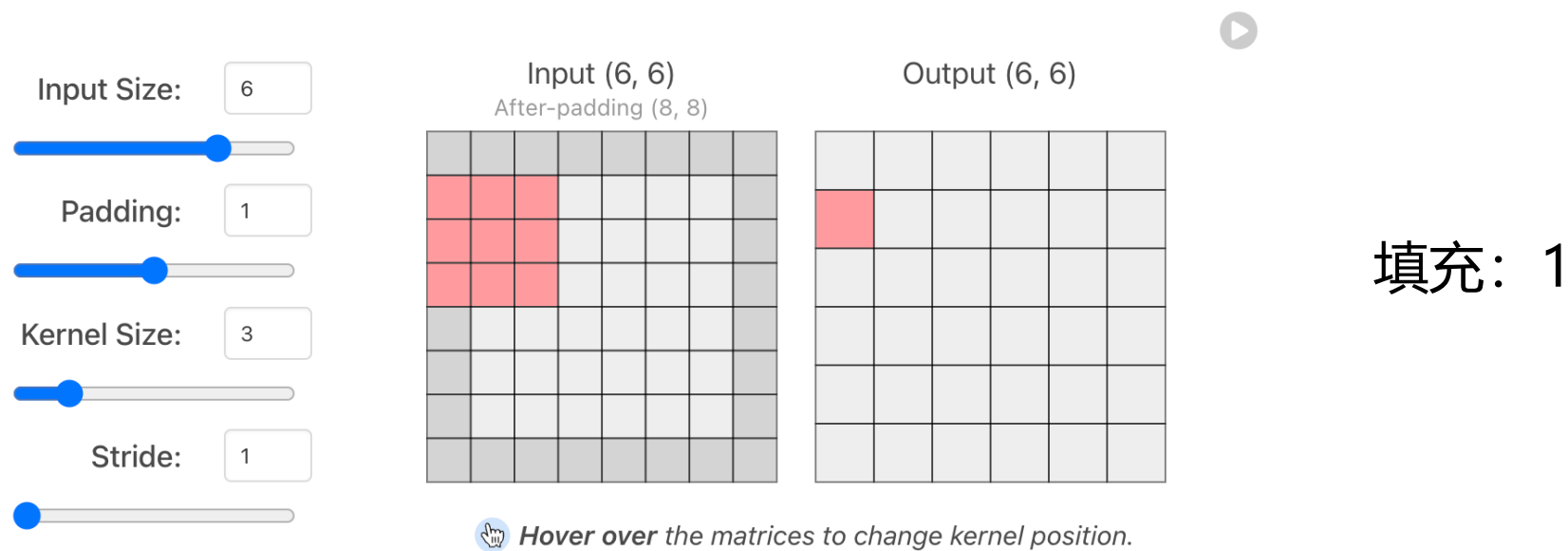
- **卷积核大小 (Kernel size)**：核大小（通常也称为滤波器大小）是指输入上方的滑动窗口的尺寸。选择此超参数会对图像分类任务产生巨大影响。例如，小核大小能够从输入中提取大量包含高度局部特征的信息。正如您在上面的可视化中看到的那样，较小的内核大小也会导致层维度的减少较小，从而允许更深入的架构。相反，较大的内核大小提取的信息较少，这会导致层维度的减少速度更快，这通常会导致性能变差。大内核更适合提取较大的特征。归根结底，选择合适的内核大小将取决于您的任务和数据集，但通常，较小的内核大小会带来更好的图像分类任务性能。

2.2 CNN的基本结构：卷积核



- **步长 (Stride)**：核一次应移动多少个像素。例如，如上面的卷积层示例所述，Tiny VGG 对其卷积层使用步幅 1，这意味着点积在输入的 3x3 窗口上执行以产生输出值，然后对于每个后续操作，点积都会向右移动一个像素。步幅对 CNN 的影响类似于内核大小。随着步幅的减小，由于提取的数据更多，会学习更多的特征，这也会导致更大的输出层。相反，随着步幅的增加，这会导致更有限的特征提取和更小的输出层维度。架构设计人员的一项职责是确保在实现 CNN 时内核对称地滑过 Importing。

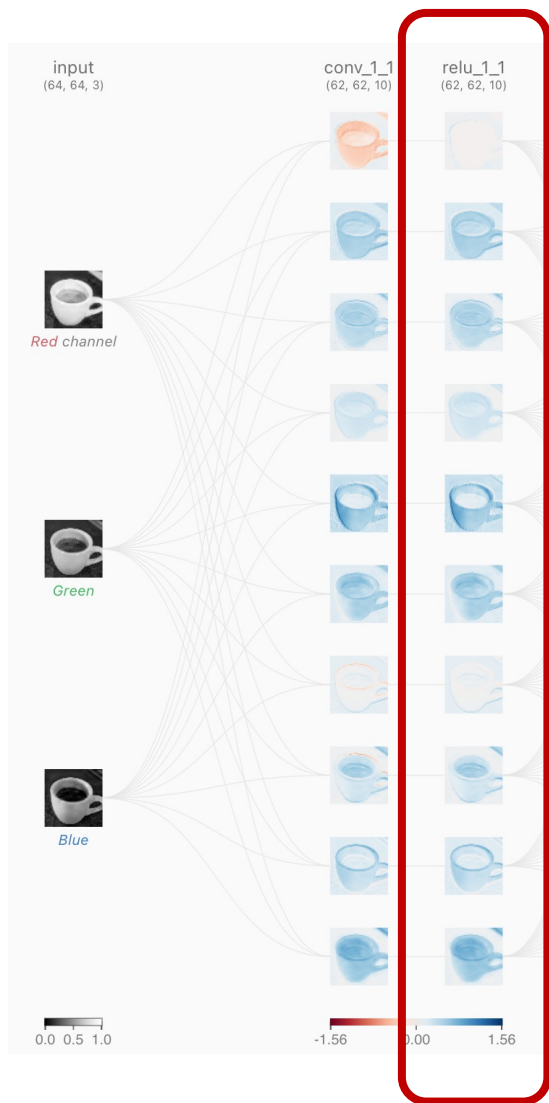
2.2 CNN的基本结构：卷积核



- **填充 (Padding)**：当核超出激活映射时，通常需要填充。填充可以保存激活映射边界处的数据，从而提高性能，并且有助于保持输入的空间大小，从而允许架构设计人员构建更深入、性能更高的网络。存在许多填充技术，但最常用的方法是零填充，因为它的性能、简单性和计算效率。该技术涉及在 input 的边缘周围对称添加 0。许多高性能 CNN（如 AlexNet）都采用这种方法。

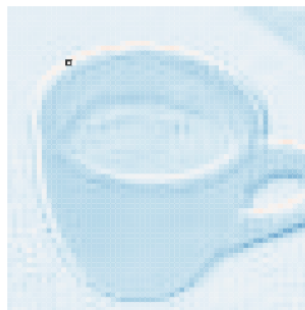
2.3 CNN的基本结构：激活函数

- 引入非线性因素
- 一般使用ReLU函数



ReLU Activation ReLU 激活

Input (62, 62) 输入 (62, 62)



$$\max(0, -0.02) =$$

最大 (,) =

0

Output (62, 62) 输出 (62, 62)

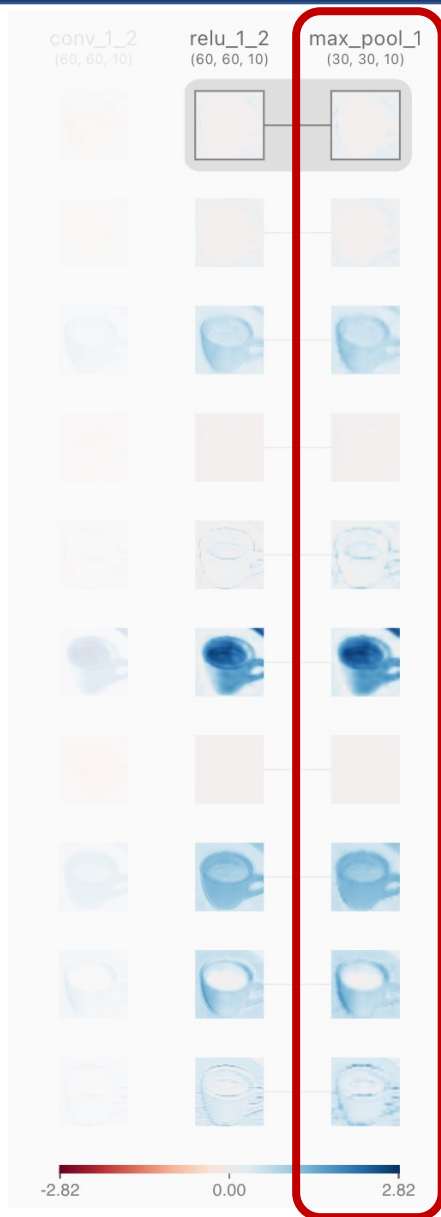


Hover over the matrices to change pixel.

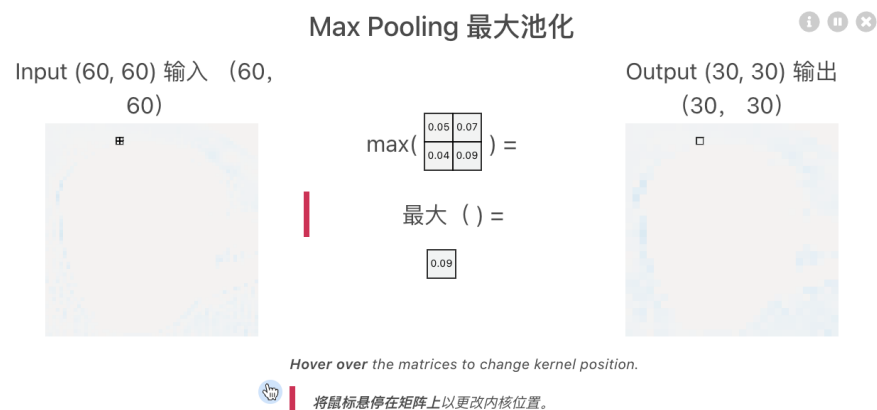


将鼠标悬停在矩阵上以更改像素。

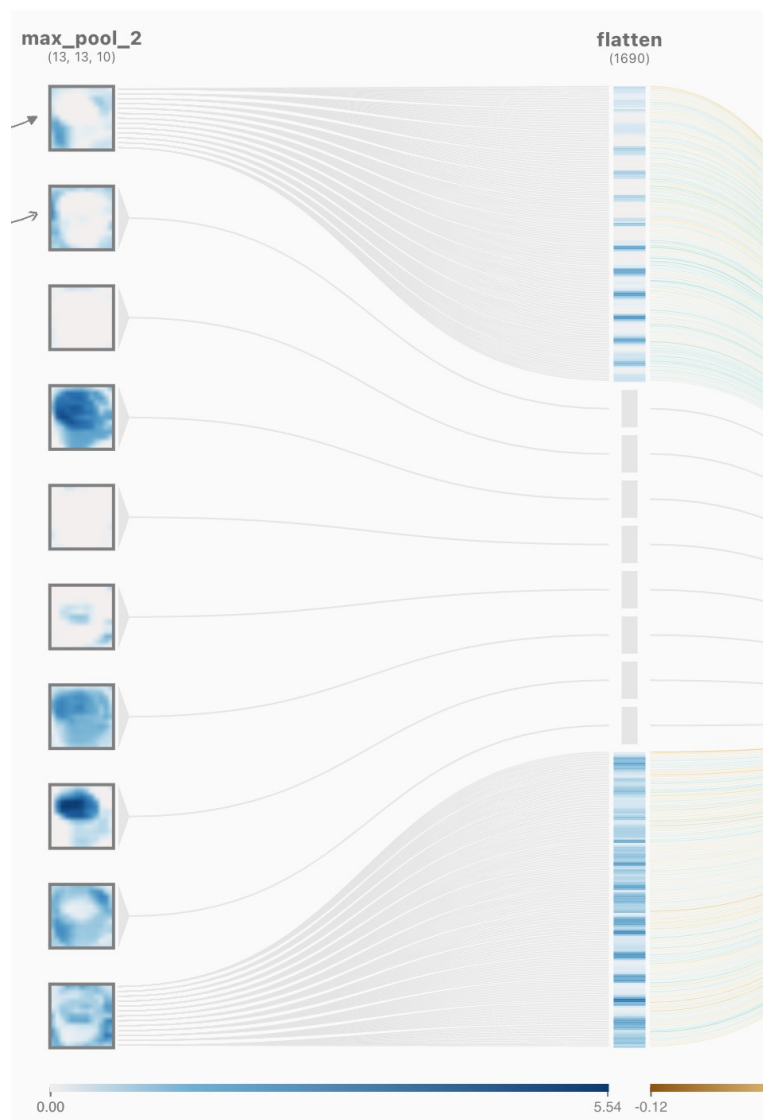
2.4 CNN的基本结构：池化层



- 不同的 CNN 架构中有许多类型的池化层，但它们的目的是逐渐减小网络的空间范围，从而减少网络的参数和整体计算。图中使用的池类型是 Max-Pooling。
- Max-Pooling 操作需要在架构设计期间选择内核大小和步长。选择后，该操作将以指定的步幅在输入上滑动内核，同时仅从输入中选择每个内核切片的最大值，以生成输出值。这个过程可以通过单击上面网络中的池化神经元来查看。
- 图中池化层使用 2x2 内核和 stride 2。具有这些规范的此操作会导致丢弃 75% 的激活。通过丢弃如此多的值，计算效率更高，并避免了过度拟合。

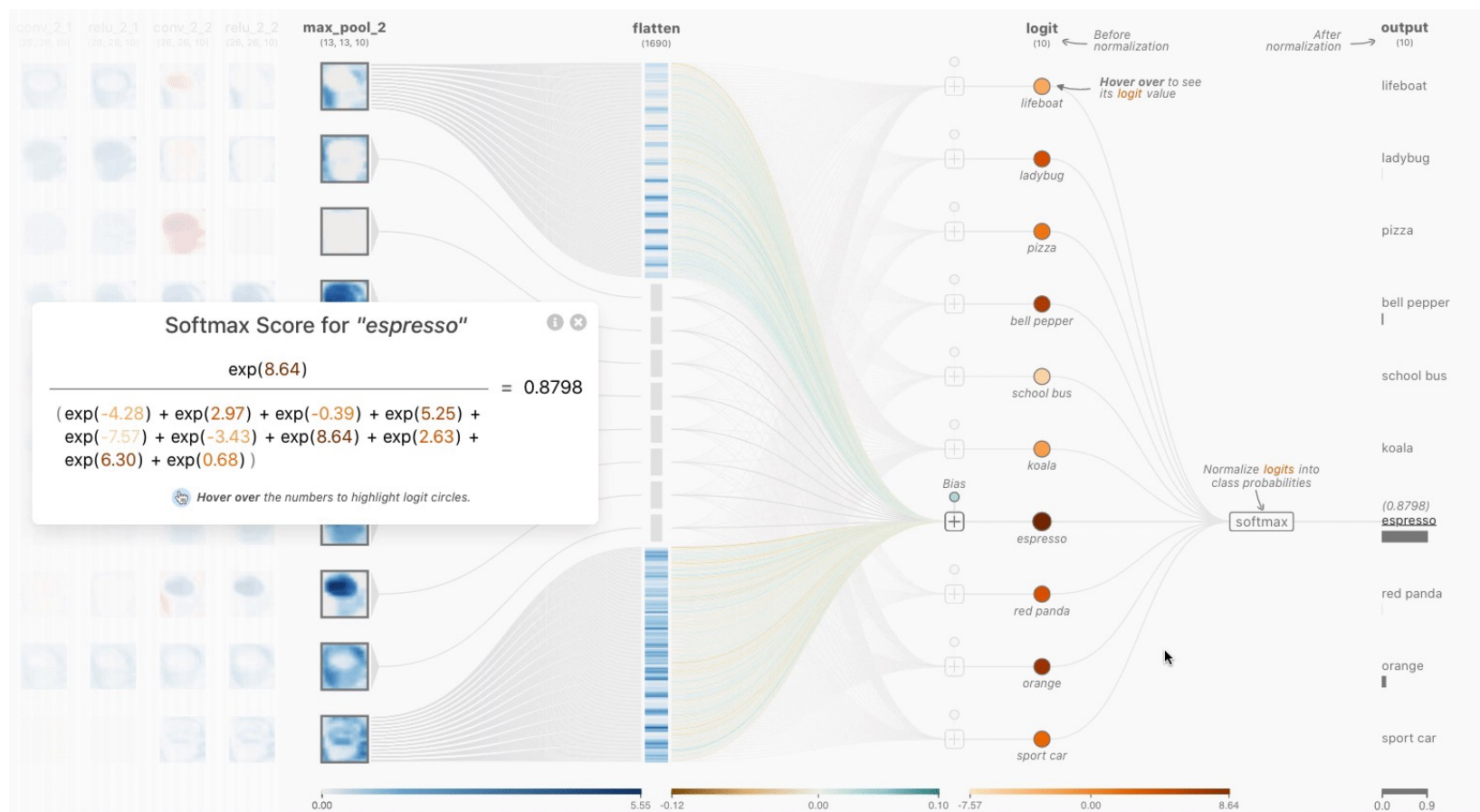


2.5 CNN的基本结构：展平层



- 该层将网络中的三维层转换为一维向量，以适应全连接层的输入进行分类。例如，一个 $5 \times 5 \times 2$ 的张量将被转换为一个大小为 50 的向量。网络的先前卷积层从输入图像中提取了特征，但现在是时候对特征进行分类了。后续输出层使用 softmax 函数对这些特征进行分类，这需要一维输入。这就是为什么需要展平层的原因。可以通过单击任何输出类来查看此图层。

2.3 CNN的基本结构：输出层



➤ 作为多元分类问题，输出层里还有一个softmax激活函数

$$\text{Softmax}(x_i) = \frac{\exp(x_i)}{\sum_j \exp(x_j)}$$

3.1 CNN的优势

- **自动特征提取**：CNN能够通过卷积层自动学习并提取数据中的关键特征，无需人工设计特征。这使得CNN在图像和视频等复杂数据处理任务中非常强大。
- **参数共享与权重共享**：卷积核在整个输入数据上共享权重，减少了网络中需要训练的参数数量。这不仅降低了计算复杂度，还增强了模型的泛化能力。
- **局部连接**：每个神经元只与输入数据的一小部分区域（局部感受野）连接，避免了全连接网络中的大量冗余连接，提高了计算效率。
- **平移不变性**：由于卷积操作的特性，CNN具有一定的平移不变性，即能够识别图像中物体的位置变化，使得CNN在图像分类、目标检测等任务中表现出色。
- **高效的层次化特征学习**：CNN通过多层结构（卷积层、池化层、全连接层）逐步提取从简单到复杂的特征，这种层次化学习有助于捕捉输入数据中的高层次结构信息。
- **广泛的应用领域**：CNN在图像分类、目标检测、语音识别、视频分析、自然语言处理等多个领域取得了显著成果，成为计算机视觉和深度学习的主流模型。

3.2 CNN的挑战

- **计算资源需求高：** 尽管CNN在特征学习上具有优势，但由于需要处理大量的卷积操作和多层参数，训练深度CNN模型需要大量的计算资源和高性能硬件（如GPU）。
- **训练时间长：** 深度CNN模型通常包含大量的参数和层次，训练过程中需要大量的时间来进行前向传播、反向传播和参数更新，尤其是在数据集较大时。
- **对大数据和标签数据的依赖：** CNN的训练通常依赖大量的标注数据来实现高性能，这对数据的收集和标签化要求较高。没有足够的高质量数据，模型的性能可能会受到影响。
- **过拟合问题：** 深度CNN模型具有强大的学习能力，但如果训练数据不足或模型过于复杂，容易发生过拟合，即模型在训练数据上表现良好，但在未见过的数据上性能差。
- **模型解释性差** CNN通常被视为“黑箱”模型，尽管其在多种任务中表现出色，但很难解释模型是如何做出决策的。这对于一些需要高解释性和透明度的应用场景（如医学影像分析）可能是一个挑战。
- **对小样本学习的适应性差：** CNN在大数据集上表现良好，但在数据样本较少的情况下（即小样本学习），其性能可能不如其他算法（如迁移学习、增强学习等）。
- **设计复杂的网络架构：** 尽管CNN在许多任务中表现优异，但设计一个合适的网络架构（如选择卷积核大小、层数、激活函数等）仍然是一个挑战，需要丰富的经验和反复的实验。

4. 练习

1. 利用Pytorch写一个用于图片分类的神经网络，包括前馈神经网络和卷积神经网络。

➤ [./03_pytorch_computer_vision.html](#)

2. 前期训练均使用其他人整理好的数据，如何准备自己的数据集。

➤ [./04_pytorch_custom_datasets.html](#)

gitee上均有原始的jupyterlab文档