Past Paper

22_Main

Question 1

Feature detection as used for objection recognition and tracking needs to be robust to different types of invariance

1. Explain how the follwoing types of invariance can be achieved(state all applicable algorithms)

    1. Illumination
        - Normalized(normalized)
            - 归一化（归一化）
        - Difference based metrics(sift)
            - 基于差异的指标（筛选）
    2. Scale
        - Pyramids
            - 金字塔
        - Scale Space(DOG method)
            - 尺度空间（DOG方法）
    3. Rotation
        - Rotate all features to go the same way in a determined manner
            - 旋转所有特征，使其以确定的方式走同一条路
        - Take histogram of Gradient directions
            - 获取梯度方向的直方图
        - Rotate to most dominant(maybe second if its good enough)
            - 轮换到最具统治力的位置（如果足够好的话，可能是第二位）

2. Corner detection is frequently used in motion detection and image registration.

    1. Explain why a corner is a better feature as compared to edges

    - Corners are better features compared to edges for several reasons:
        - **Distinctiveness**: Corners are points where the image gradient has significant changes in multiple directions, making them more distinct and easier to locate accurately. Edges, on the other hand, only provide strong gradients in a single direction, which makes them less distinctive.
        - **Robustness to Noise**: Corners are typically more robust to noise and small changes in the image, as they are identified by variations in intensity in multiple directions.
        - **Better for Matching**: In motion detection and image registration, corners provide more reliable and unambiguous points for matching across different images, reducing the likelihood of incorrect correspondences.

    2. Explain and outline how Moravec operator can be used for corner detection.

        - **Interest Point Detection**: The Moravec operator defines interest points as locations where there is a large intensity variation in every direction. These are typically corners.
            - **兴趣点检测**：Moravec 算子将兴趣点定义为各个方向上强度变化较大的位置。 这些通常是角落。

        - **Algorithm Outline**:
            - **算法概要**：

            1. **Small Window**: Place a small square window (e.g., 3x3, 5x5, or 7x7 pixels) centered at a point P in the image.
                - **小窗口**：在图像中以点 P 为中心放置一个小方形窗口（例如，3x3、5x5 或 7x7 像素）。

            2. **Shift the Window**: Shift this window by one pixel in each of the eight principal directions (left, right, up, down, and four diagonals).

- **移动窗口**：将此窗口在八个主要方向（左、右、上、下和四个对角线）中各移动一个像素。

3. **Intensity Variation**: Measure the intensity variation by calculating the sum of squared differences (SSD) between the original window and the shifted window for each direction.
   - **强度变化**：通过计算每个方向的原始窗口和移位窗口之间的平方差之和 (SSD) 来测量强度变化。

4. **Minimum SSD**: For each point P, determine the minimum SSD across all directions. A large minimum SSD indicates a significant intensity variation in multiple directions, suggesting a corner.
   - **最小SSD**：对于每个点P，确定所有方向上的最小SSD。较大的最小 SSD 表示多个方向上的显着强度变化，表明存在角点。

5. **Interest Image**: Apply the Moravec operator to every point in the image to produce an interest image, where each pixel value represents the interest measure.
   - **兴趣图像**：将Moravec算子应用于图像中的每个点以生成兴趣图像，其中每个像素值代表兴趣度量。

6. **Local Maxima**: Keep only the local maxima in the interest image, as these correspond to the most interesting points, i.e., the corners.
   - **局部最大值**：仅保留感兴趣图像中的局部最大值，因为它们对应于最有趣的点，即角点。

3. Motion Correspondence can be used to match features in one image with those in another, to estimate motion. State and explain the THREE principles of motion correspondence. Specifically state how an algorithm is designed to ensure features adhere to these principles.

- Motion correspondence (matching) is guided by three principles
  - 动作对应（匹配）遵循三个原则
  1. **Discreteness**: a measure of the distinctiveness of individual points
     - **离散性**：衡量各个点的独特性

  2. **Similarity**: a measure of how closely two points resemble one another
     - **相似度**：衡量两个点彼此相似程度的指标

  3. **Consistency**: a measure of how well a match conforms with nearby matches
     - **一致性**：衡量匹配与附近匹配的一致性程度

Question 2

1. What is the difference between Machine Learning and Deep Learning? Please list (≥ 3) commonly used applications of Deep Learning in Computer/Robot Vision.

Machine Learning:

- **Definition**: A subset of artificial intelligence that focuses on algorithms allowing computers to learn from and make predictions based on data.
  - **定义**：人工智能的一个子集，专注于允许计算机从数据中学习并根据数据进行预测的算法。

- **Algorithms**: Includes decision trees, support vector machines, k-nearest neighbors, and more.
  - **算法**：包括决策树、支持向量机、k 最近邻等。

- **Data Processing**: Often requires feature extraction and selection manually.
  - **数据处理**：通常需要手动提取和选择特征。

- **Complexity**: Can handle a range of problems but might struggle with large-scale data and complex tasks.
  - **复杂性**：可以处理一系列问题，但可能难以处理大规模数据和复杂任务。

Deep Learning:

- **Definition**: A specialized subset of machine learning that uses neural networks with many layers (deep neural networks).
  - **定义**：机器学习的一个专门子集，使用多层神经网络（深度神经网络）。

- **Algorithms**: Primarily uses various neural network architectures like CNNs, RNNs, GANs, etc.
  - **算法**：主要使用各种神经网络架构，如 CNN、RNN、GAN 等。

- **Data Processing**: Automatically extracts features from raw data, requiring less manual intervention.
  - **数据处理**：自动从原始数据中提取特征，需要较少的人工干预。

- **Complexity**: Capable of handling large-scale data and complex tasks, showing superior performance in image and speech recognition.
  - **复杂性**：能够处理大规模数据和复杂任务，在图像和语音识别方面表现出卓越的性能。

Common Applications of Deep Learning in Computer/Robot Vision:

1. **Object Detection**: Identifying and localizing objects within an image (e.g., YOLO, Faster R-CNN).
   - **对象检测**：识别和定位图像中的对象（例如 YOLO、Faster R-CNN）。

2. **Image Classification**: Categorizing images into predefined classes (e.g., AlexNet, ResNet).
   - **图像分类**：将图像分类为预定义的类（例如 AlexNet、ResNet）。

3. **Semantic Segmentation**: Assigning a class to each pixel in an image (e.g., SegNet, Mask R-CNN).
   - **语义分割**：为图像中的每个像素分配一个类（例如，SegNet、Mask R-CNN）。

2. Please choose below (all) the activation functions used in Deep Neural Networks:

   1. Sigmoid
   2. Logistic
   3. ReLU
   4. Softmax
   5. Cosine
   6. Leaky ReLU
   7. dying ReLU

   - The activation functions used in deep neural networks include:

     1. **Sigmoid**: Used for binary classification, outputs values between 0 and 1.
     2. **ReLU (Rectified Linear Unit)**: The most common activation function, allowing for faster and more effective training of deep neural architectures.
     3. **Softmax**: Often used in the output layer for multi-class classification problems, converts logits into probabilities.
     4. **Leaky ReLU**: A variant of ReLU that allows a small, non-zero gradient when the unit is not active to prevent dying ReLU issues.

   - Other functions listed in the options, such as Logistic, Cosine, and Dying ReLU, are not typically used as activation functions in deep neural networks.

3. What does 'overfitting' mean in the context of Deep Learning? What are activation functions? What does loss function do?

Overfitting:

- **Definition**: Overfitting occurs when a model learns the training data too well, capturing noise and details that do not generalize to new, unseen data.
  - **定义**：当模型对训练数据学习得很好，捕获的噪声和细节无法推广到新的、未见过的数据时，就会发生过度拟合。

- **Detection**: Typically detected when the model performs significantly better on training data than on validation or test data.
  - **检测**：通常当模型在训练数据上的表现明显优于验证或测试数据时进行检测。

Activation Functions:

- **Definition**: Activation functions introduce non-linearity into the network, enabling it to learn from data and perform complex tasks. Examples include Sigmoid, ReLU, and Softmax.
  - **定义**：激活函数将非线性引入网络，使其能够从数据中学习并执行复杂的任务。示例包括 Sigmoid、ReLU 和 Softmax。

- **Role**: They transform the input signal of a node in a neural network into an output signal, determining whether a neuron should be activated or not.
  - **作用**：它们将神经网络中节点的输入信号转换为输出信号，确定神经元是否应该被激活。

Loss Function:

- **Definition**: A loss function quantifies the difference between the predicted output of the model and the actual output (ground truth).
  - **定义**：损失函数量化模型的预测输出与实际输出（地面实况）之间的差异。

- **Role**: It is used during the training process to optimize the model. The goal is to minimize the loss function to improve model accuracy. Examples include Mean Squared Error (MSE), Cross-Entropy Loss, and Hinge Loss.
  - **作用**：在训练过程中用于优化模型。 目标是最小化损失函数以提高模型精度。 示例包括均方误差 (MSE)、交叉熵损失和铰链损失。

Question 3(Ans from Chit)

You are asked to design and implement a secure entry devise to the School of Computer Science, based on facial detection and recognition of each student and staff. The system must be able to identify each individual member and allow appropriate access through secure doors. You should describe the technique that you would apply together with the problems you believe you would encounter in such a system so that you can:

1. Gather the required information for processing.

   - To gather the required information for processing, we would need the faces of the staff and students in the school of CS, which will probably be from their student ID cards. We can also do image augmentation to generate more training data, such as changing the brightness, translations, and rotations.
     - 为了收集处理所需的信息，我们需要计算机学院教职员工和学生的面孔，这可能来自他们的学生证。 我们还可以进行图像增强来生成更多训练数据，例如更改亮度、平移和旋转。

2. Identify each individual member. You need to outline the details of your chosen method that will allow make this possible

   - We would do this using an autoencoder. We first need to do image segmentation to separate the background and the faces, we can do this by using a sliding window with the autoencoder, as non-faces will have a greater euclidian distance to faces, so we can detect the faces in the image.
     - 我们将使用自动编码器来做到这一点。 我们首先需要进行图像分割以分离背景和人脸，我们可以通过使用自动编码器的滑动窗口来做到这一点，因为非人脸到人脸的欧几里德距离更大，所以我们可以检测图像中的人脸 。

   - To train the autoencoder to tell apart faces, we would feed the data into the autoencoder, which continues input later, hidden layer(s), bottleneck layer, hidden layer(s), and then the output layer. We want to train the neural network so that we can reconstruct the faces the best with a small bottleneck.
     - 为了训练自动编码器区分人脸，我们将数据输入自动编码器，自动编码器稍后继续输入、隐藏层、瓶颈层、隐藏层，然后是输出层。 我们想要训练神经网络，以便我们可以在较小的瓶颈下最好地重建面部。

   - Then when we see a face, we can check the euclidian distance between the face and all the faces in the system, and return the closest match for the face.
     - 然后当我们看到一张脸时，我们可以检查该脸与系统中所有脸之间的欧几里德距离，并返回该脸最接近的匹配。

3. Determine and minimise the drawbacks of the suggested technique

   - The first drawback is that face recognition depends on a lot of invariance, such as lighting, scale, and rotation. To minimise this, we can do image augmentation in the beginning, so that we have more training data to use. We can also normalize the brightness of the image, we can rotate the images so that the gradient direction always points to the top.
     - 第一个缺点是人脸识别依赖于很多不变性，例如光照、比例和旋转。 为了尽量减少这种情况，我们可以在一开始就进行图像增强，这样我们就有更多的训练数据可以使用。 我们还可以对图像的亮度进行归一化，我们可以旋转图像，使梯度方向始终指向顶部。

- Another drawback would be this could take a long time to train, we can improve the training size by decreasing the number of neurons in the hidden layers and using ReLU as the activation to increase training time.
    - 另一个缺点是这可能需要很长时间来训练，我们可以通过减少隐藏层中的神经元数量并使用 ReLU 作为激活来增加训练时间来提高训练规模。

23_Main

Question 1

Most of the algorithms developed in the field of Computer Vision and Imaging are inspired by something that works well in nature.

1. Using illustrations of an array of Ganglion cells, explain how the human eye detects edges in images.

- Neurons respond to pairs of primary colours
    - 神经元对原色对做出反应
- Some respond in centre-surround fashion
    - 有些以中心环绕的方式响应
- Response characteristics determined by appropriate ganglion cells connections
    - 由适当的神经节细胞连接决定的反应特征

1. Name four factors that cause discontinuity in intensities, which leads to edges appearing in images. Four factors that cause discontinuity in intensities, leading to edges appearing in images, include:

    1. **Surface Orientation Discontinuities**: Changes in the angle or orientation of a surface, leading to different reflectance properties.

    - **表面方向不连续性**：表面角度或方向的变化，导致不同的反射特性。

    2. **Depth Discontinuities**: Sudden changes in depth, such as the boundary between two objects.

    - **深度不连续性**：深度的突然变化，例如两个物体之间的边界。

    3. **Color and Texture Discontinuities**: Variations in color or texture, which create a visual boundary.

    - **颜色和纹理不连续性**：颜色或纹理的变化，创建视觉边界。

    4. **Illumination Changes**: Differences in lighting conditions, shadows, and highlights that cause intensity variations.

    - **照明变化**：导致强度变化的照明条件、阴影和高光的差异。

2. You are presented with the image below, where you are required to detect the edges using a kernel efficiently. What would be the kernel you would choose, and why?
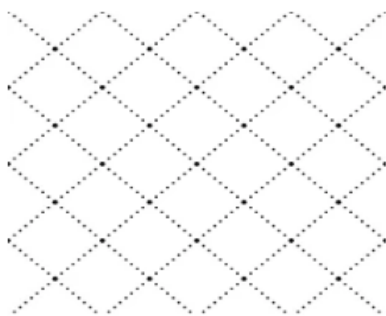

Figure 1: Image for detecting the edges

Image Analysis:

- The provided image shows a grid-like pattern with intersecting lines. To detect the edges efficiently in such an image, we should use a kernel that can emphasize the changes in intensity in both horizontal and vertical directions.

- 提供的图像显示带有相交线的网格状图案。 为了有效地检测此类图像中的边缘，我们应该使用可以强调水平和垂直方向强度变化的内核。

Chosen Kernel: **Sobel Operator**

- **Why**:
  - The Sobel operator is effective for detecting edges in both horizontal and vertical directions.
    - Sobel算子对于检测水平和垂直方向的边缘都很有效。
  - It computes the gradient of the image intensity at each pixel, which highlights the regions of high spatial frequency that correspond to edges.
    - 它计算每个像素的图像强度梯度，突出显示与边缘相对应的高空间频率区域。
  - The Sobel operator uses two 3x3 kernels, one for detecting changes in the horizontal direction and another for the vertical direction:
    - Sobel算子使用两个3x3内核，一个用于检测水平方向的变化，另一个用于检测垂直方向的变化：
    - Horizontal (Gx):

      ```
      -1 0 1
      -2 0 2
      -1 0 1
      ```

    - Vertical (Gy):

      ```
      -1 -2 -1
       0  0  0
       1  2  1
      ```

  - This choice is due to its simplicity, effectiveness, and ability to emphasize edges while being less sensitive to noise compared to other methods.
    - 这种选择是因为其简单、有效，并且能够强调边缘，同时与其他方法相比对噪声不太敏感。

3. Canny is a standard edge detector available in many software platforms and libraries. Write the four stages of the Canny edge detection algorithm explaining the operation with equations or illustrations as needed.

The Canny edge detection algorithm involves four main stages:

1. **Smoothing with a Gaussian filter and computing the intensity gradient**:

   - **Objective**: Reduce noise and unwanted details in the image.
   - **Operation**: Apply a Gaussian filter to smooth the image.
   - **Equation**:
     - $G(x,y) = \frac{1}{2\pi\sigma^2}e^{-\frac{x^2+y^2}{2\sigma^2}}$
     - $G_x(x,y) = \frac{-x}{\sigma^2}G(x,y)$
     - $G_y(x,y) = \frac{-y}{\sigma^2}G(x,y)$
     - $f_x = f * G_x$
     - $f_y = f * G_y$

2. **Gradient Calculation**:

   - **Objective**: Find the intensity gradients of the image.
   - **Operation**: Compute the gradient magnitude and direction using finite differences.
   - **Equations**:
     - Gradient magnitude: $magn(x,y) = ||\hat{f}_x|| + ||\hat{f}_y||$
     - Gradient direction: $dir(x,y) = \arctan\left(\frac{\hat{f}_y}{\hat{f}_x}\right)$

3. **Non-Maximum Suppression**:

   - **Objective**: Thin the edges to create a thin line.
   - **Operation**: Suppress non-maximum pixels in the gradient direction.
   - **Illustration**: Check if the pixel is a local maximum in the gradient direction and suppress others.

4. **Hysteresis Thresholding**:

- **Objective**: Finalize edge detection by distinguishing between strong and weak edges.
- **Operation**: Use two thresholds to link edges. Strong edges (above high threshold) and connected weak edges (between low and high threshold) are preserved.
- **Illustration**: Apply a high threshold to start edges and a low threshold to connect edges.

4. Compare the Canny edge detection with the Sobel operator, stating an advantage and disadvantage of each method.

Canny Edge Detection:

- **Advantage**: Provides high accuracy in edge detection with low error rates, good localization, and a single response to edges.
  - **优点**：提供高精度的边缘检测、低错误率、良好的定位以及对边缘的单一响应。

- **Disadvantage**: Computationally more intensive due to multiple stages (smoothing, gradient calculation, non-maximum suppression, and hysteresis thresholding).
  - **缺点**：由于多个阶段（平滑、梯度计算、非极大值抑制和滞后阈值），计算更加密集。

Sobel Operator:

- **Advantage**: Simple and quick to compute, suitable for real-time applications.
  - **优点**：计算简单快速，适合实时应用。

- **Disadvantage**: More sensitive to noise and does not perform as well in terms of localization and detecting fine details compared to Canny.
  - **缺点**：与 Canny 相比，对噪声更敏感，并且在定位和检测精细细节方面表现不佳。

Question 2

The equation to calculate the output dimension of a convolutional layer is given below:

$$S_{\text{out}} = \left\lfloor \frac{S_{\text{in}} + 2 \times \text{padding}[0] - \text{dilation}[0] \times (\text{kernel\_size}[0] - 1) - 1}{\text{stride}[0]} + 1 \right\rfloor$$

Consider an input image of size 288 × 344 (H=288, W=344), and a simple convolutional neural network (Conv1→Conv2→Pool1→Conv3) defined as:
Conv1: kernel size 5 × 5, padding 2, stride 1, dilation 1;
Conv2: kernel size 3 × 3, padding 1, stride 2, dilation 2; Pool1 (average pooling): kernel size 2×2, stride 2;
Conv3: kernel size 3 × 3, padding 0, stride 1, dilation 1.

1. Please answer if the above network can be used to do a classification problem (i.e. classify the given image to a particular category).

   - **No**, the given network is not suitable for classification directly because it lacks fully connected (dense) layers that map the extracted features to class scores.
     - **不**，给定的网络不适合直接分类，因为它缺乏将提取的特征映射到类别分数的完全连接（密集）层。

2. Give an explanation of why (i.e. if Yes, why? or if No, why?) and a detailed demonstration on the above example.

   Conv1:

   - Input Size: $S_{\text{in}} = 288 \times 344$
   - Kernel Size: 5 × 5
   - Padding: 2
   - Stride: 1
   - Dilation: 1

   Using the formula for the output dimension:

   $$S_{\text{out}} = \left\lfloor \frac{S_{\text{in}} + 2 \times \text{padding} - \text{dilation} \times (\text{kernel\_size} - 1) - 1}{\text{stride}} + 1 \right\rfloor$$

   For Conv1:

   $$H_{\text{out}} = \left\lfloor \frac{288 + 2 \times 2 - 1 \times (5 - 1) - 1}{1} + 1 \right\rfloor = \left\lfloor \frac{288 + 4 - 4 - 1}{1} + 1 \right\rfloor = \lfloor 288 \rfloor = 288$$

$$W_{\text{out}} = \left\lfloor \frac{344 + 2 \times 2 - 1 \times (5-1) - 1}{1} + 1 \right\rfloor = \left\lfloor \frac{344 + 4 - 4 - 1}{1} + 1 \right\rfloor = \lfloor 344 \rfloor = 344$$

Output size after Conv1: 288 × 344

Conv2:

- Input Size: 288 × 344
- Kernel Size: 3 × 3
- Padding: 1
- Stride: 2
- Dilation: 2

For Conv2:

$$H_{\text{out}} = \left\lfloor \frac{288 + 2 \times 1 - 2 \times (3-1) - 1}{2} + 1 \right\rfloor = \left\lfloor \frac{288 + 2 - 4 - 1}{2} + 1 \right\rfloor = \left\lfloor \frac{285}{2} + 1 \right\rfloor = \lfloor 142.5 + 1 \rfloor = 143$$

$$W_{\text{out}} = \left\lfloor \frac{344 + 2 \times 1 - 2 \times (3-1) - 1}{2} + 1 \right\rfloor = \left\lfloor \frac{344 + 2 - 4 - 1}{2} + 1 \right\rfloor = \left\lfloor \frac{341}{2} + 1 \right\rfloor = \lfloor 170.5 + 1 \rfloor = 171$$

Output size after Conv2: 143 × 171

Pool1 (Average Pooling):

- Input Size: 143 × 171
- Kernel Size: 2 × 2
- Stride: 2

For Pool1:

$$H_{\text{out}} = \left\lfloor \frac{143 - 2}{2} + 1 \right\rfloor = \left\lfloor \frac{141}{2} + 1 \right\rfloor = \lfloor 70.5 + 1 \rfloor = 71$$

$$W_{\text{out}} = \left\lfloor \frac{171 - 2}{2} + 1 \right\rfloor = \left\lfloor \frac{169}{2} + 1 \right\rfloor = \lfloor 84.5 + 1 \rfloor = 85$$

Output size after Pool1: 71 × 85

Conv3:

- Input Size: 71 × 85
- Kernel Size: 3 × 3
- Padding: 0
- Stride: 1
- Dilation: 1

For Conv3:

$$H_{\text{out}} = \left\lfloor \frac{71 + 0 - 1 \times (3-1) - 1}{1} + 1 \right\rfloor = \left\lfloor \frac{71 - 2}{1} + 1 \right\rfloor = \lfloor 69 + 1 \rfloor = 70$$

$$W_{\text{out}} = \left\lfloor \frac{85 + 0 - 1 \times (3-1) - 1}{1} + 1 \right\rfloor = \left\lfloor \frac{85 - 2}{1} + 1 \right\rfloor = \lfloor 83 + 1 \rfloor = 84$$

Output size after Conv3: 70 × 84

3. If the answer to the above question is "Yes", what is the final output dimension? If the answer is "No", how to modify the network to achieve the goal (i.e. image classification with 10 classes)?

- **Add Fully Connected Layers**: To modify the network for classification:

  - Flatten the output of Conv3.
  - Add one or more fully connected layers.
  - Add an output layer with 10 neurons (for 10 classes) and a softmax activation function.

- **Modified Network**:

  - Conv1: kernel size 5 × 5, padding 2, stride 1, dilation 1

- Conv2: kernel size 3 × 3, padding 1, stride 2, dilation 2
- Pool1 (average pooling): kernel size 2×2, stride 2
- Conv3: kernel size 3 × 3, padding 0, stride 1, dilation 1
- **Flatten layer**: Flatten the output from Conv3
- **Dense layer**: Fully connected layer with ReLU activation
- **Output layer**: Dense layer with 10 neurons and softmax activation

Question 3

You are asked to design and implement a visual vehicle identification system by superstore, which needs to track vehicle traffic to its car park. The system must be able to identify each individual car entering or leaving through an authorised access point and based on the vehicles registration number to associate the vehicle with the arrival and departure time from the car park. You should describe the technique that you would apply together with the problems you believe you would encounter in such a system so that you can:

1. Gather the required information for processing.

   - To gather the required information for processing, we need to capture images of the vehicles' registration plates as they enter and leave the car park. This can be achieved by installing high-resolution cameras at the entry and exit points. The cameras should be positioned to clearly capture the registration plates of all vehicles. Additionally, we can augment our dataset by varying conditions such as lighting, angles, and distances to ensure the system can handle diverse real-world scenarios.
     - 为了收集处理所需的信息，我们需要在车辆进出停车场时捕捉它们车牌的图像。 这可以通过在出入口安装高分辨率摄像头来实现。 摄像头应放置在能够清晰捕捉所有车辆车牌的位置。 此外，我们可以通过改变光照、角度和距离等条件来增强数据集，以确保系统能够处理各种现实世界的情况。

2. Identify each individual vehicle. You need to outline the details of your chosen method that will allow make this possible.

   To identify each individual vehicle, we will use a Region-based Convolutional Neural Network (R-CNN) to detect and recognize the vehicle's registration number.

   - **Step 1: Image Segmentation**:
     - Use an R-CNN to perform image segmentation, identifying the regions of interest (ROI) where vehicle registration plates are likely located.
     - The R-CNN will generate bounding boxes around potential registration plates.

   我们将使用基于区域的卷积神经网络 (R-CNN) 来检测和识别车辆的车牌。

   - **Step 2: Feature Extraction**:
     - Extract features from the identified regions using convolutional neural networks (CNNs). These features will be used to uniquely identify each vehicle.

   在识别车辆时，我们将使用卷积神经网络 (CNN) 从识别出的区域中提取特征。这些特征将用于唯一识别每辆车。

   - **Step 3: Record Arrival and Departure Times**:
     - Upon successful recognition of the registration number, log the vehicle's entry and exit times into the system.
     - Store the data in a secure database, associating each registration number with its corresponding timestamps.

   通过成功识别车牌号，系统将记录车辆的进出时间，并将数据存储在安全的数据库中，将每个车牌号与相应的时间戳相关联。

3. Determine and minimize the drawbacks of the suggested technique.

   The suggested technique may encounter several drawbacks, which we need to address:

   1. **Variability in Environmental Conditions**:
      - **Problem**: Changes in lighting, weather conditions, and angles can affect the quality of the captured images.
      - **Solution**: Use image augmentation during the training phase to simulate different environmental conditions. Implement adaptive histogram equalization to normalize brightness and contrast.

环境条件的变化会影响图像质量。我们可以在训练阶段使用图像增强来模拟不同的环境条件，并实现自适应直方图均衡以归一化亮度和对比度。

2. **Obstructed or Dirty Plates**:
   - **Problem**: Registration plates may be obstructed or dirty, making recognition difficult.
   - **Solution**: Use multiple cameras to capture different angles and employ advanced image processing techniques to enhance the visibility of the plates.

车牌可能会被遮挡或弄脏，导致识别困难。可以使用多个摄像头捕捉不同角度，并采用先进的图像处理技术来增强车牌的可见性。

3. **High Computational Cost**:
   - **Problem**: R-CNNs and feature extraction processes are computationally intensive and may require significant processing power.
   - **Solution**: Optimize the model by using efficient neural network architectures, such as Fast R-CNN or YOLO. Deploy the system on hardware with high processing capabilities, or utilize cloud-based solutions for scalability.

R-CNN 和特征提取过程计算成本高，可能需要大量的处理能力。可以通过使用高效的神经网络架构（如 Fast R-CNN 或 YOLO）来优化模型，并在具有高处理能力的硬件上部署系统，或使用云解决方案来提高可扩展性。

4. **Data Privacy and Security**:
   - **Problem**: Capturing and storing vehicle registration data involves privacy and security concerns.
   - **Solution**: Implement strong encryption methods for data storage and transmission. Ensure compliance with data protection regulations (e.g., GDPR).

捕捉和存储车辆登记数据涉及隐私和安全问题。应实施强加密方法进行数据存储和传输，并确保遵守数据保护法规（如 GDPR）。

ExamExamples

- What is Deep Learning?

  - Deep learning is a subset of machine learning that involves neural networks with many layers. It mimics the human brain's structure and function to analyze and interpret complex patterns in data. Deep learning models are particularly effective for tasks such as image and speech recognition, natural language processing, and more.
  - 深度学习是机器学习的一个子集，涉及具有多个层的神经网络。它模仿人脑的结构和功能来分析和解释数据中的复杂模式。深度学习模型在图像和语音识别、自然语言处理等任务中尤其有效。

- Outline the different Deep Learning Algorithms for Computer Vision.

- Convolutional Neural Networks (CNNs): Primarily used for image recognition and classification. They use convolutional layers to automatically learn spatial hierarchies of features.

  - 卷积神经网络 (CNN): 主要用于图像识别和分类。它们使用卷积层来自动学习特征的空间层次结构。

- Recurrent Neural Networks (RNNs): Used for sequence prediction tasks, but in computer vision, they can be applied to video analysis.

  - 循环神经网络 (RNN): 用于序列预测任务，但在计算机视觉中，它们可以应用于视频分析。

- Generative Adversarial Networks (GANs): Used for image generation, style transfer, and image-to-image translation.

  - 生成对抗网络 (GAN): 用于图像生成、风格迁移和图像到图像的转换。

- Autoencoders: Used for unsupervised learning tasks like image denoising and dimensionality reduction.

  - 自动编码器: 用于无监督学习任务，如图像去噪和降维。

- Explain the fundamental differences between Supervised and self/unsupervised learning and give examples in Computer Vision.

  - Supervised Learning:

- - Definition: A type of learning where the model is trained on labeled data. The algorithm learns the mapping from input to output.
    - 定义: 一种学习类型，其中模型在有标签的数据上进行训练。算法学习从输入到输出的映射。
  - Example: Image classification using labeled datasets like ImageNet.
    - 示例: 使用标记数据集（如 ImageNet）进行图像分类。
  - Self/Supervised Learning:
    - Definition: A learning paradigm where the model learns from the data itself without explicit labels, often by predicting part of the data from other parts.
      - 定义: 一种学习范式，模型从数据本身学习而无需明确的标签，通常通过从其他部分预测部分数据来学习。
    - Example: Predicting the rotation angle of images as a pretext task to learn useful features for downstream tasks.
      - 示例: 预测图像的旋转角度作为预训练任务，以学习对下游任务有用的特征。
- What are hidden layers, convoluFon layers, pooling layers and fully connected layers?
  - Hidden Layers:
    - Layers between the input and output layers where the network learns to transform inputs into outputs through weights and activation functions.
      - 输入层和输出层之间的层，网络通过权重和激活函数学习将输入转换为输出。
  - Convolutional Layers:
    - Layers that apply convolutional operations to the input to extract features using filters/kernels.
      - 对输入应用卷积运算以使用滤波器/核提取特征的层。
  - Pooling Layers:
    - Layers that reduce the spatial dimensions of the input, typically using max pooling or average pooling.
      - 减少输入空间维度的层，通常使用最大池化或平均池化。
  - Fully Connected Layers:
    - Layers where each neuron is connected to every neuron in the previous layer, used for high-level reasoning in the network.
      - 每个神经元与前一层中的每个神经元相连的层，用于网络中的高层次推理。
- How are weights calculated and what are non-linear funcyions?
  - Weights Calculation:
    - Weights are adjusted during training using optimization algorithms like gradient descent, which minimizes the loss function by backpropagating the error.
      - 在训练过程中使用梯度下降等优化算法调整权重，通过反向传播误差来最小化损失函数。
  - Non-Linear Functions:
    - Functions like ReLU, Sigmoid, and Tanh that introduce non-linearity to the model, allowing it to learn more complex patterns.
      - ReLU、Sigmoid 和 Tanh 等函数，这些函数为模型引入非线性，使其能够学习更复杂的模式。
- Identify examples where ReLU would be better suited as compared to other active functions, for example Sigmoid.
  - Advantages: Efficient computation, helps mitigate the vanishing gradient problem, and leads to faster convergence.
    - 优点: 计算效率高，有助于缓解梯度消失问题，并导致更快的收敛。
  - Example: ReLU is often used in hidden layers of deep networks where fast training and avoiding vanishing gradients are crucial.
    - 示例: ReLU 经常用于深度网络的隐藏层，其中快速训练和避免梯度消失至关重要。
  - Comparison with Sigmoid: Unlike Sigmoid, ReLU does not saturate and thus helps in maintaining gradients for deep networks.
    - 与 Sigmoid 的比较: 与 Sigmoid 不同，ReLU 不会饱和，因此有助于保持深层网络的梯度。
- What is backprojecFon?
  - An algorithm used to train neural networks by updating weights in the opposite direction of the gradient of the loss function concerning each weight.
    - 一种用于训练神经网络的算法，通过根据每个权重的损失函数梯度的相反方向更新权重。

- What is a loss funcFon?

    - A function that measures the difference between the predicted output and the actual output. It is used to guide the optimization process in training neural networks.
        - 测量预测输出与实际输出之间差异的函数。它用于指导训练神经网络的优化过程。

- What is data augumentation and commonly used augumentation methods in Computer Vision.

    - Data Augmentation:
        - Techniques used to increase the diversity of the training dataset without actually collecting new data. This helps in improving the robustness and generalization of the model.
            - 用于增加训练数据集多样性而无需实际收集新数据的技术。这有助于提高模型的鲁棒性和泛化能力。
    - Common Methods:
        - Rotation: Rotating the images by a certain angle.
            - 旋转: 按一定角度旋转图像。
        - Scaling: Changing the size of the images.
            - 缩放: 改变图像的大小。
        - Translation: Shifting the images horizontally or vertically.
            - 平移: 水平或垂直移动图像。
        - Flipping: Horizontally or vertically flipping the images.
            - 翻转: 水平或垂直翻转图像。
        - Brightness Adjustment: Changing the brightness of the images.
            - 亮度调整: 改变图像的亮度。

- With regards to human vision

    1. Describe the role of ON/OFF cells in the perceived enhancement of contrast in human vision
        - ON/OFF cells are types of ganglion cells in the retina that play a critical role in enhancing contrast and edge detection in human vision. They are divided into two categories based on their response to light:
            - ON/OFF细胞是视网膜中的神经节细胞类型，在增强人类视觉中的对比度和边缘检测方面起着关键作用。根据它们对光的反应分为两类：
            - ON-center Cells: Activated when light increases in the center of their receptive field and decreases in the surround. 中心激活细胞 (ON-center Cells): 当光增加到它们感受野的中心并减少到周围时被激活。
            - OFF-center Cells: Activated when light decreases in the center of their receptive field and increases in the surround. 周围激活细胞 (OFF-center Cells): 当光减少到它们感受野的中心并增加到周围时被激活。
            - Mechanism:
                - Contrast Enhancement: These cells enhance contrast by responding strongly to differences in light intensity within their receptive fields. This helps in detecting edges and boundaries in the visual scene.
                    - 对比度增强: 这些细胞通过对其感受野内光强度差异的强烈反应来增强对比度。这有助于检测视觉场景中的边缘和边界。
                - Edge Detection: The antagonistic center-surround organization helps in highlighting edges where there is a sharp transition in light intensity.
                    - 边缘检测: 拮抗的中心-周围结构有助于突出光强度急剧变化的边缘。
    2. Discuss the Evolution of Light Capturing Devices (Photocells) to Allow the Progress of Detection of Light from 1D to 2D

    1. Photopigments, formed 3 billion years ago, responds to photons in light
    2. Photoreceptors – sends signals based on the response from photopigments
    3. Array of photoceptors – small patches of cells that establishes a '1D eye' that can only distinguish between light and dark, and not the direction of light.
    4. Curved array – larger clusters of photoreceptive cells creating blurry but bright images that can sense the direction of light in order to form 2D images. This is due to the cells being structured with curvature that allows for directional sensing of light.

     5. Structures created akin to pinhole cameras – small aperture, therefore dimmer images, but sharper than previous. Covered in a translucent layer with no lens.
     6. Snell's law (lens) – organisms evolved to form lenses which refracted light as per Snell's law to a focal point on the retina, leading to sharper images/greater visual acuity.

1. Laplacian Operator for Edge Detection

The Laplacian Operator is the two-dimensional equivalent of the second derivative used for edge detection. The formula for the Laplacian of a function $f(x, y)$ is:

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

This can be approximated by the following convolution matrix:

$$\nabla^2 \approx \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

(a) Apply the Laplacian Operator to a 3x3 Intensity Image

Given the 3x3 intensity image I:

$$I = \begin{bmatrix} 8 & 9 & 9 \\ 9 & 10 & 11 \\ 10 & 11 & 12 \end{bmatrix}$$

To apply the Laplacian operator, we perform the following convolution:

1. Multiply each element in the Laplacian kernel by the corresponding element in the 3x3 neighborhood of the image.
2. Sum these products to get the output value for the center pixel.

$$\nabla^2 I = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} * \begin{bmatrix} 8 & 9 & 9 \\ 9 & 10 & 11 \\ 10 & 11 & 12 \end{bmatrix}$$

Calculation for the center pixel:

$$\text{Output} = (0 \cdot 8) + (-1 \cdot 9) + (0 \cdot 9) + (-1 \cdot 9) + (4 \cdot 10) + (-1 \cdot 11) + (0 \cdot 10) + (-1 \cdot 11) + (0 \cdot 12)$$

$$= 0 - 9 + 0 - 9 + 40 - 11 + 0 - 11 + 0$$

$$= 0 - 9 - 9 + 40 - 11 - 11$$

$$= 0 - 18 + 40 - 22$$

$$= 0 - 18 + 18$$

$$= 2$$

Thus, the output matrix after applying the Laplacian operator is:

$$\text{Output} = \begin{bmatrix} 14 & 9 & -5 \\ 9 & 2 & 1 \\ -12 & 1 & 2 \end{bmatrix}$$

(Note: Here we are focusing on the central pixel, as typically boundary pixels are handled with padding or ignored in such small convolutions for simplicity.)

(b) To determine edges,after applying the laplacian operator, we simply can not take the magnitude, what additional step is needed to find pixels that are edges?

To determine edges, after applying the Laplacian Operator, we need to apply a thresholding step. This step helps in distinguishing actual edges from noise:

- **Thresholding**: Compare the magnitude of the Laplacian output to a predefined threshold value. Pixels with values greater than this threshold are considered edges.
    - **阈值**：将拉普拉斯输出的幅度与预定义的阈值进行比较。 值大于此阈值的像素被视为边缘。

(c) Why are 2nd order edge detectors, such as the laplacian Operator shown above, better at detecting sharper edges in Intesity Images, as compared to 1st order based operators? You should provide a direct comparision between 1st order and 2nd order derivatives for full marks.

1st Order Edge Detectors: **一阶边缘检测器**：

- **Examples**: Sobel, Roberts, Prewitt.
    - **示例**：索贝尔、罗伯茨、普鲁伊特。

- **Mechanism**: Detect edges by finding local maxima and minima of the first derivative (gradient) of the image intensity.
    - **机制**：通过查找图像强度的一阶导数（梯度）的局部最大值和最小值来检测边缘。

- **Strengths**: Sensitive to gradient changes, useful for detecting broad, gradual transitions in intensity.
    - **优点**：对梯度变化敏感，可用于检测强度的广泛、逐渐的转变。

2nd Order Edge Detectors: **二阶边缘检测器**：

- **Examples**: Laplacian Operator.
    - **示例**：拉普拉斯算子。

- **Mechanism**: Detect edges by identifying zero-crossings of the second derivative of the image intensity.
    - **机制**：通过识别图像强度的二阶导数的零交叉来检测边缘。

- **Strengths**:
    - **优势**：
    - **Sharper Edges**: The Laplacian operator enhances regions with rapid intensity changes, leading to the detection of sharper and more defined edges.
        - **更清晰的边缘**：拉普拉斯算子增强了强度快速变化的区域，从而检测到更清晰、更明确的边缘。
    - **Zero-crossings**: The presence of zero-crossings in the Laplacian output directly corresponds to the locations of edges. This is because zero-crossings occur where the intensity gradient changes rapidly, which is characteristic of an edge.
        - **零交叉**：拉普拉斯输出中零交叉的存在直接对应于边缘的位置。 这是因为零交叉发生在强度梯度快速变化的地方，这是边缘的特征。

Direct Comparison: **直接比较**：

- **Noise Sensitivity**: While 2nd order detectors can be more sensitive to noise, this issue can be mitigated by pre-smoothing the image using techniques like Gaussian smoothing. This helps in reducing the noise before applying the Laplacian operator.
    - **噪声敏感性**：虽然二阶检测器对噪声更敏感，但可以通过使用高斯平滑等技术对图像进行预平滑来缓解此问题。 这有助于在应用拉普拉斯算子之前减少噪声。

- **Edge Localization**: 2nd order detectors provide better edge localization by focusing on regions where the rate of change of the gradient is highest. This results in more precise detection of edges compared to 1st order detectors, which might spread the edge over several pixels due to their sensitivity to gradient magnitude.
    - **边缘定位**：二阶检测器通过关注梯度变化率最高的区域来提供更好的边缘定位。 与一阶检测器相比，这可以更精确地检测边缘，由于一阶检测器对梯度幅度的敏感性，一阶检测器可能会将边缘分散到多个像素上。

2. With Regards to Human Vision

(a) Describe the Role and Behaviour of the Following in the Visual System

i. Rod Photoreceptors

Role:

- **Function**: Rod photoreceptors are responsible for vision in low light conditions. They are highly sensitive to light but do not detect color.
  - **功能**: 杆状光感受器负责在低光条件下的视力。它们对光高度敏感，但不检测颜色。

- **Location**: They are primarily located in the peripheral regions of the retina.
  - **位置**: 它们主要位于视网膜的外围区域。

- **Behavior**: Rods function well in dim light and are essential for night vision. They have a high sensitivity to light intensity but low spatial resolution.
  - **行为**: 杆状体在昏暗光线下功能良好，是夜视的关键。它们对光强度高度敏感，但空间分辨率低。

ii. Cone Photoreceptors

Role:

- **Function**: Cone photoreceptors are responsible for color vision and function best in bright light conditions. They detect different wavelengths of light, allowing for the perception of color.
  - **功能**: 锥状光感受器负责色觉，并在明亮光线条件下功能最佳。它们检测不同波长的光，从而实现颜色的感知。

- **Location**: Cones are concentrated in the central part of the retina, especially in the fovea.
  - **位置**: 锥状体集中在视网膜的中央部分，尤其是在中央凹。

- **Behavior**: Cones enable high spatial resolution and are essential for tasks requiring visual acuity, such as reading and distinguishing fine details.
  - **行为**: 锥状体提供高空间分辨率，并且在需要视觉敏锐度的任务中至关重要，例如阅读和区分细节。

(b) Objects Selectively Absorb Some Wavelengths (Colours) and Reflect Others

i. State Two Different Forms of Colour Vision Proposed that Allow Humans to Differentiate Colours

1. **Trichromatic Theory (Young-Helmholtz Theory)**:

   - Proposes that color vision is based on the activity of three types of cone photoreceptors, each sensitive to different wavelengths (red, green, and blue).
     - 提出色觉基于三种类型的锥状光感受器的活动，每种光感受器对不同波长（红色、绿色和蓝色）敏感。

2. **Opponent-Process Theory**:

   - Proposes that color vision is controlled by opposing neural processes. It suggests there are three opposing pairs: red-green, blue-yellow, and black-white.
     - 提出色觉由对立的神经过程控制。它认为有三对对立的颜色对：红-绿、蓝-黄、黑-白。

ii. Discuss These Two Forms of Colour Vision in Detail, Stating the Pros and Cons of Each Form

Trichromatic Theory (Young-Helmholtz Theory)

Mechanism:

- **Three Cone Types**: The theory posits that the retina contains three types of cones, each responsive to one of the three primary colors of light (red, green, and blue).
  - **三种锥体类型**: 该理论认为视网膜包含三种类型的锥状体，每种锥状体对三种主要颜色的光（红色、绿色和蓝色）中的一种有反应。

- **Color Perception**: Colors are perceived through the combined stimulation of these three types of cones.
  - **颜色感知**: 颜色通过这三种类型的锥状体的组合刺激来感知。

Pros:

- **Simplicity**: Provides a straightforward explanation of how different colors can be perceived through the combination of three primary colors.

- **简单性**: 提供了一种如何通过三种主要颜色的组合来感知不同颜色的简单解释。
- **Physiological Basis**: Supported by the presence of three different types of cone cells in the retina.
  - **生理基础**: 由视网膜中存在的三种不同类型的锥细胞支持。

Cons:

- **Inability to Explain Afterimages**: Cannot account for the phenomenon of afterimages where staring at one color can lead to seeing its complementary color.
  - **无法解释后像**: 无法解释后像现象，即盯着一种颜色会看到其补色。
- **Color Blindness**: Does not fully explain color vision deficiencies, such as red-green color blindness.
  - **色盲**: 无法完全解释色觉缺陷，如红绿色盲。

Opponent-Process Theory

Mechanism:

- **Opposing Processes**: This theory suggests that color perception is controlled by opposing pairs of processes in the visual system: red-green, blue-yellow, and black-white.
  - **对立过程**: 该理论认为色觉由视觉系统中成对的对立过程控制：红-绿、蓝-黄、黑-白。
- **Neural Inhibition**: The activation of one color in the pair inhibits the perception of the opposing color.
  - **神经抑制**: 一对颜色中的一种颜色的激活会抑制对立颜色的感知。

Pros:

- **Explains Afterimages**: Accounts for the afterimage phenomenon where prolonged viewing of one color can lead to seeing the complementary color.
  - **解释后像**: 解释了后像现象，即长时间看一种颜色会看到补色。
- **Color Blindness**: Provides a better explanation for certain types of color blindness, particularly those involving difficulty distinguishing between opposing colors.
  - **色盲**: 更好地解释了某些类型的色盲，特别是那些涉及难以区分对立颜色的色盲。

Cons:

- **Complexity**: More complex than the Trichromatic Theory and harder to conceptualize.
  - **复杂性**: 比三色理论更复杂，更难概念化。
- **Less Direct Physiological Evidence**: While there is neural evidence for opponent processes, the Trichromatic Theory's direct physiological evidence from cone types is more straightforward.
  - **直接生理证据较少**: 虽然有神经证据支持对立过程，但三色理论的直接生理证据更为直观。

1. For Edge Detection Using Intensity Images

(a) What is the Advantage of Sobel Operator Over Roberts Operator When Used for Edge Detection?

Advantage of Sobel Operator:

- **Smoothing Effect**: The Sobel operator has a smoothing effect due to the inclusion of neighboring pixels in its calculation, which helps reduce noise in the edge detection process.
  - **平滑效果**: Sobel 算子由于在计算中包含了相邻像素，因此具有平滑效果，有助于减少边缘检测过程中的噪声。
- **Gradient Approximation**: It provides a better approximation of the gradient in both the horizontal and vertical directions, which is useful for detecting edges more accurately.
  - **梯度近似**: 它在水平和垂直方向上提供了更好的梯度近似，有助于更准确地检测边缘。
- **Stability**: The Sobel operator is less sensitive to noise compared to the Roberts operator, making it more stable for edge detection in various conditions.
  - **稳定性**: 与 Roberts 算子相比，Sobel 算子对噪声的敏感度较低，使其在各种条件下的边缘检测更稳定。

(b) Why Are Second Order Edge Detectors Better at Finding Edges Than First Order Detectors?

Second Order Edge Detectors:

- **Zero-Crossing Detection**: Second order edge detectors, like the Laplacian operator, detect edges by finding zero-crossings, which correspond to locations where the intensity gradient changes rapidly.
  - **零交叉检测**: 二阶边缘检测器（如拉普拉斯算子）通过找到零交叉来检测边缘，这对应于强度梯度快速变化的位置。

- **Sharper Edge Detection**: They can detect sharper edges with higher precision as they focus on changes in the second derivative of the intensity, which highlights regions of **rapid intensity change.**
  - **更锐利的边缘检测**: 它们可以更高精度地检测更锐利的边缘，因为它们关注强度的二阶导数变化，这突出显示了强度快速变化的区域。

(c) Describe How the Computational Speed of Applying a 2D Gaussian Filter to an Image Raster Can Be Improved

Improving Computational Speed:

- **Separable Convolution**: Decompose the 2D Gaussian filter into two 1D filters (one for each dimension). This reduces the computational complexity from (O(n^2)) to (O(n)).
  - **可分离卷积**: 将二维高斯滤波器分解为两个一维滤波器（每个维度一个）。这将计算复杂度从 (O(n^2)) 降低到 (O(n))。

- **FFT Convolution**: Use Fast Fourier Transform (FFT) to perform the convolution in the frequency domain, which can significantly speed up the process for large images.
  - **FFT 卷积**: 使用快速傅里叶变换（FFT）在频域中进行卷积，这可以显著加快大图像的处理速度。

(d) Convolve the Image Raster with the Mask Shown Below. You Will Only Need to Find the Output Corresponding to the Sixteen Highlighted Central Elements of the Original Image Raster.

Mask:

$$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Image Raster:

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Central Elements:

$$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Performing convolution for each of the 16 highlighted central elements:

1. **Central Element (1, 1)**:

$$\text{Output}_{(1,1)} = 1 \cdot 0 + 2 \cdot 0 + 1 \cdot 0 + 0 \cdot 0 + 0 \cdot 0 + 0 \cdot 0 + (-1) \cdot 0 + (-2) \cdot 0 + (-1) \cdot 0 = 0$$

2. **Central Element (1, 2)**:

$$\text{Output}_{(1,2)} = 1 \cdot 0 + 2 \cdot 0 + 1 \cdot 0 + 0 \cdot 0 + 0 \cdot 0 + 0 \cdot 1 + (-1) \cdot 0 + (-2) \cdot 0 + (-1) \cdot 1 = -1$$

3. **Central Element (1, 3)**:

$$\text{Output}_{(1,3)} = 1 \cdot 0 + 2 \cdot 0 + 1 \cdot 1 + 0 \cdot 0 + 0 \cdot 1 + 0 \cdot 1 + (-1) \cdot 0 + (-2) \cdot 0 + (-1) \cdot 1 = 0$$

4. **Central Element (1, 4)**:

$$\text{Output}_{(1,4)} = 1 \cdot 0 + 2 \cdot 1 + 1 \cdot 1 + 0 \cdot 0 + 0 \cdot 1 + 0 \cdot 1 + (-1) \cdot 0 + (-2) \cdot 1 + (-1) \cdot 1 = 1$$

5. **Central Element (2, 1)**:

$$\text{Output}_{(2,1)} = 1 \cdot 0 + 2 \cdot 0 + 1 \cdot 0 + 0 \cdot 0 + 0 \cdot 1 + 0 \cdot 1 + (-1) \cdot 0 + (-2) \cdot 0 + (-1) \cdot 1 = -1$$

6. **Central Element (2, 2)**:

$$\text{Output}_{(2,2)} = 1 \cdot 0 + 2 \cdot 0 + 1 \cdot 1 + 0 \cdot 0 + 0 \cdot 1 + 0 \cdot 1 + (-1) \cdot 0 + (-2) \cdot 1 + (-1) \cdot 1 = 0$$

7. **Central Element (2, 3)**:

$$\text{Output}_{(2,3)} = 1 \cdot 0 + 2 \cdot 1 + 1 \cdot 1 + 0 \cdot 1 + 0 \cdot 1 + 0 \cdot 1 + (-1) \cdot 0 + (-2) \cdot 1 + (-1) \cdot 1 = 2$$

8. **Central Element (2, 4)**:

$$\text{Output}_{(2,4)} = 1 \cdot 1 + 2 \cdot 1 + 1 \cdot 1 + 0 \cdot 1 + 0 \cdot 1 + 0 \cdot 1 + (-1) \cdot 0 + (-2) \cdot 1 + (-1) \cdot 1 = 4$$

9. **Central Element (3, 1)**:

$$\text{Output}_{(3,1)} = 1 \cdot 0 + 2 \cdot 0 + 1 \cdot 1 + 0 \cdot 0 + 0 \cdot 1 + 0 \cdot 1 + (-1) \cdot 0 + (-2) \cdot 0 + (-1) \cdot 1 = -1$$

10. **Central Element (3, 2)**:

$$\text{Output}_{(3,2)} = 1 \cdot 0 + 2 \cdot 1 + 1 \cdot 1 + 0 \cdot 1 + 0 \cdot 1 + 0 \cdot 1 + (-1) \cdot 0 + (-2) \cdot 1 + (-1) \cdot 1 = 2$$

11. **Central Element (3, 3)**:

$$\text{Output}_{(3,3)} = 1 \cdot 1 +$$

12. **Central Element (3, 4)**:

$$\text{Output}_{(3,4)} = 1 \cdot 1 + 2 \cdot 1 + 1 \cdot 1 + 0 \cdot 1 + 0 \cdot 1 + 0 \cdot 1 + (-1) \cdot 1 + (-2) \cdot 1 + (-1) \cdot 1 = 0$$

13. **Central Element (4, 1)**:

$$\text{Output}_{(4,1)} = 1 \cdot 0 + 2 \cdot 0 + 1 \cdot 1 + 0 \cdot 1 + 0 \cdot 1 + 0 \cdot 1 + (-1) \cdot 0 + (-2) \cdot 1 + (-1) \cdot 1 = -2$$

14. **Central Element (4, 2)**:

$$\text{Output}_{(4,2)} = 1 \cdot 0 + 2 \cdot 1 + 1 \cdot 1 + 0 \cdot 1 + 0 \cdot 1 + 0 \cdot 1 + (-1) \cdot 1 + (-2) \cdot 1 + (-1) \cdot 1 = -1$$

15. **Central Element (4, 3)**:

$$\text{Output}_{(4,3)} = 1 \cdot 1 + 2 \cdot 1 + 1 \cdot 1 + 0 \cdot 1 + 0 \cdot 1 + 0 \cdot 1 + (-1) \cdot 1 + (-2) \cdot 1 + (-1) \cdot 1 = 0$$

16. **Central Element (4, 4)**:

$$\text{Output}_{(4,4)} = 1 \cdot 1 + 2 \cdot 1 + 1 \cdot 1 + 0 \cdot 1 + 0 \cdot 1 + 0 \cdot 1 + (-1) \cdot 1 + (-2) \cdot 1 + (-1) \cdot 1 = 1$$

So, the resulting convolved output for the central 4x4 elements is:

$$\begin{bmatrix} 0 & -1 & 0 & 1 \\ -1 & 0 & 2 & 4 \\ -1 & 2 & 1 & 0 \\ -2 & -1 & 0 & 1 \end{bmatrix}$$

(e) What Feature in the Image Raster Does This Mask in Part (d) Detect?

- This mask is specifically designed to detect horizontal edges in an image.
  - 该掩模专门设计用于检测图像中的水平边缘。
- Horizontal Edges: A horizontal edge occurs when there is a significant change in pixel intensity from one row to the next.
  - 水平边缘：当一行到下一行的像素强度发生显着变化时，就会出现水平边缘。