

程式的使用方法：

```
1  #include <iostream>
2  #include <fstream>
3
問題 5 輸出 偵錯主控台 終端機 連接埠 GITLENS
● jxnhd@CHANDeMacBook-Pro ~/Desktop/c++final [master] g++ -std=c++11 main.cpp -o t
○ jxnhd@CHANDeMacBook-Pro ~/Desktop/c++final [master] ./t
Please input your file name (ppm file)
█
```

首先要輸入程式的名字.ppm

```
問題 5 輸出 偵錯主控台 終端機 連接埠 GITLENS
● jxnhd@CHANDeMacBook-Pro ~/Desktop/c++final [master] g++ -std=c++11 main.cpp -o t
○ jxnhd@CHANDeMacBook-Pro ~/Desktop/c++final [master] ./t
Please input your file name (ppm file)
input_image.ppm
Filter system
1. Boxfilter filter
2. Identity filter
3. Median filter
Please choose the filter █
```

第二步，請選擇你想要的過濾器

```
問題 5 輸出 偵錯主控台 終端機 連接埠 GITLENS
● jxnhd@CHANDeMacBook-Pro ~/Desktop/c++final [master] g++ -std=c++11 main.cpp -o t
○ jxnhd@CHANDeMacBook-Pro ~/Desktop/c++final [master] ./t
Please input your file name (ppm file)
input_image.ppm
Filter system
1. Boxfilter filter
2. Identity filter
3. Median filter
Please choose the filter 3
Please input the size(3, 5, 7)
█
```

以中值濾鏡為例（盒式濾鏡同樣操作），輸入尺寸（3*3，5*5，7*7），然後就會輸出處理過的 ppm 檔案。

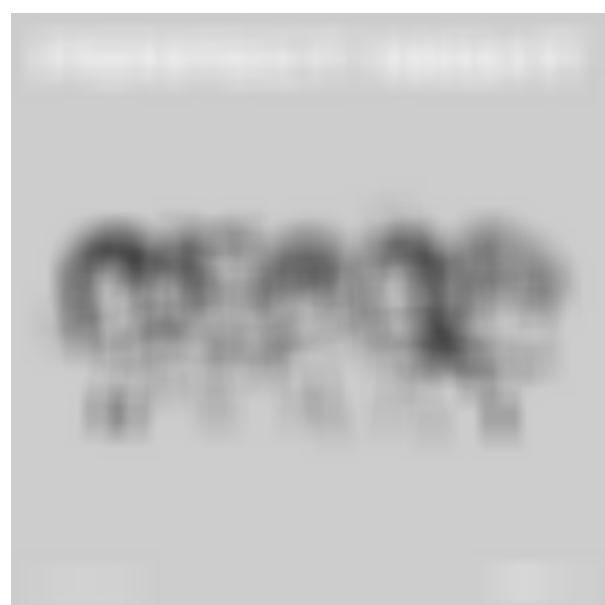
```
Please input your file name (ppm file)
input_image.ppm
Filter system
1. Boxfilter filter
2. Identity filter
3. Median filter
Please choose the filter 3
Please input the size(3, 5, 7)
3
the image is successfully output
```

輸入的圖片：



（提外話，他們的歌挺好聽）
以這張圖片為例（以轉換 ppm 檔案）

盒式濾鏡 (3*3, 5*5, 7*7, 31*31)



恆等濾鏡



中值濾鏡 (3X3, 5X5, 7X7)



結果分析：

● 盒式濾鏡

從輸出結果可以得知，當我輸入的尺寸越來越大，我的圖片就會變得越來越模糊。從而得知，卷積核越大，圖片就會變得越來越模糊。

當卷積核的大小增加時，圖片變得模糊的現象是由於卷積核的尺寸影響了濾波過程中對周圍像素的平均化效果。這種平均化可以被視為一種模糊操作。在方框濾波器的情況下，輸出的每個像素值都是卷積核內像素值的平均。當卷積核的大小增加時，每個輸出像素都受到更多周圍像素的影響，因此整體上會導致平均化效果更加強烈。這會使得圖片中的細節和高頻信息被進一步模糊化。在實際情況中，我們可以使用這個濾鏡進行降燥以及模糊圖片

這個程式實現了一個簡單的圖像處理流程，主要分為讀取 PPM 圖片、應用方框濾波器、生成高分辨率 ASCII 藝術和寫入 PPM 文件這幾個步驟。以下是每個步驟的詳細解釋：

1. 讀取 PPM 圖片：

- 使用 **readPPM** 函數讀取 PPM 圖片。這個函數會檢查文件格式是否為 P2（ASCII 灰度）並讀取圖片的寬、高以及像素數據。
- 讀取的像素數據存儲在 **inputImage** 向量中。

2. 方框濾波器的應用：

- 使用 **boxFilter** 函數對讀取的 PPM 圖片進行方框濾波。這個函數對每個像素應用方框濾波器，計算鄰域內像素的平均值，生成濾波後的圖片，存儲在 **outputImage** 向量中。

3. 生成高分辨率 ASCII 藝術：

- 使用 **asciiArtToFile** 函數將濾波後的像素數據轉換為 ASCII 字符，並寫入文本文件。這個函數使用一組 ASCII 字符來表示不同灰度值，生成具有高分辨率的 ASCII 藝術。
- 生成的 ASCII 藝術文本文件以 **output_ascii_art_kernel_*.txt** 命名。

4. 寫入 PPM 文件：

- 使用 **writePPM** 函數將濾波後的像素數據寫入新的 PPM 文件。這個函數將像素數據轉換為 P2 格式的 PPM 文件，其中包括圖片的寬、高以及像素數據。
- 寫入的 PPM 文件以 **output_image_kernel_*.ppm** 命名。

5. 主函數：

- 在 **main** 函數中，首先讀取名為 "input_image.ppm" 的 PPM 圖

片。

- 定義了三種不同的卷積核尺寸（3、5、7和31），並對每個卷積核尺寸應用上述步驟，生成相應的高分辨率 ASCII 藝術和 PPM 文件。

● 恆等濾鏡

當使用恆等濾鏡時，可以發現輸出的結果其實和輸入沒有區別。

恆等濾鏡的工作模式非常簡單，它實際上是一個不進行任何像素變換的濾鏡。它的作用是將輸入圖像的每一個像素值都直接複製到輸出圖像中，而不對這些值進行任何修改。這使得輸出圖像保持與輸入圖像相同的外觀和內容。

在一個灰度圖像的例子中，每個像素的值表示了灰度值，範圍通常在0到255之間。恆等濾鏡通常按照以下步驟工作：

1. 對於每個像素，讀取其灰度值。
2. 直接將該灰度值複製到輸出圖像的對應位置。
3. 重複上述步驟，處理所有的像素。

這樣，輸出圖像將與輸入圖像完全相同。因此，恆等濾鏡的主要目的是保持圖像的原始信息，而不進行任何修改。這在一些情況下可能會被用作基本的濾鏡，或者作為一個比較基準，用來確保其他濾鏡的效果。

● 中值濾鏡

中值濾波（Median Filter）是一種非線性濾波技術，主要用於去除圖像中的脈衝雜訊和其他離群值。其工作原理是透過將每個像素點的灰度值替換為其周圍鄰域內像素灰度值的中值。這種方法對於脈衝雜訊有較好的去噪效果，因為中值對於異常值不敏感。

- 定義鄰域：對於每個像素，選擇一個鄰域，通常是一個正方形或矩形區域，然後取這個鄰域內的所有像素值。
- 排序：對選定的鄰域內的像素值進行排序，以便找到中間值（對於奇數大小的鄰域）或中間兩個值的平均值（對於偶數大小的鄰域）。
- 取中值：將中間值或中間兩個值的平均值分配給目標像素，從而替換原始像素值。

中值濾波是一種常用於圖像處理的非線性濾波器，主要應用於去除圖像中的脈衝雜訊。其優勢在於對於離群值的穩健性，能有效地替代異常值，使其在去噪方面表現良好。中值濾波的實現相對簡單，算法不複雜，而且對

圖像邊緣信息的保留較好，適用於一些需要維持圖像細節的場景。然而，中值濾波也存在一些缺點，例如可能導致圖像細節的模糊和計算開銷較大。

這個C++程式主要實現了中值濾波器（Median Filter）來處理輸入的PPM格式的圖像。中值濾波器是一種非線性濾波技術，適用於去除圖像中的雜訊，尤其是椒鹽噪聲。以下是這個程式的實作解釋：

1. 讀取PPM圖像：

- 程式開始時使用 **readPPM** 函數讀取輸入的PPM圖像。**readPPM** 函數會打開文件，讀取圖像的寬度、高度以及像素值，並將這些數據存儲在二維整數向量 **image** 中。

2. 實現中值濾波：

- **medianFilter** 函數負責實現中值濾波。它採用三個參數：**image** 是輸入圖像的像素數據，**width** 和 **height** 分別是圖像的寬度和高度。
- 中值濾波運算發生在 **filteredImage** 這個二維整數向量中。對於每個像素，它通過選擇指定的鄰域內的所有像素值，排序這些值，並將中值賦予目標像素，這樣就替換了原始像素值。

3. 寫入處理後的圖像：

- **writePPM** 函數負責將經過中值濾波處理的圖像寫入PPM格式的文件。它將 **filteredImage** 中的像素數據寫入文件。

4. 主函數(main)：

- **main** 函數是整個程序的入口。它首先讀取原始的PPM格式圖像，然後應用3x3、5x5和7x7的中值濾波器來處理圖像。最後，它將處理後的圖像寫入新的PPM文件中，以保存結果。

5. 程序的運行：

- 你需要將原始的PPM格式圖像保存為文件，並確保程式中的文件名匹配。程式將生成三個不同尺寸的處理後的PPM文件：**filtered_3x3.ppm**、**filtered_5x5.ppm**和**filtered_7x7.ppm**。

心得：

此作業只有我一個人一組。因我已經學習過線性代數，傅立葉轉換，所以這份較為容易理解到一些原理。我覺得這份作業的難度其實是偏難，一個人做其實工作量是偏大，因此我認為這份作業應該限制為二人一組，這樣會較好。這此作業其實就是把我之前學習的線代原理，以及在課堂上學習到的指標使用。因此當結果出來的時候我很有成就感。但因為**ascii Art**的部分我處理很久，還是不能輸出整張**ppm**圖片，因此這部分有挫折感。然後就是這份報告我使用了標題檔的技巧。因為當把所以程式放到同一份檔案，發現到代碼的行數十分多，要**debug**的時候十分痛苦，因此我只能把部分**function**放到**.h file**。這此經歷也告訴我未來進行大型程式設計，可以使用這個方法使我的程式更簡潔。