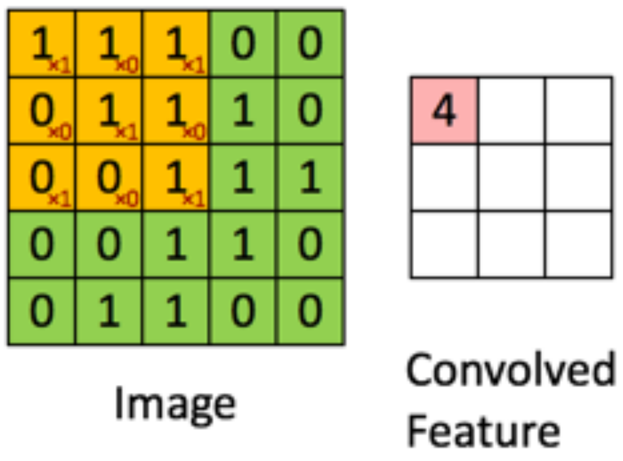
Convolution:

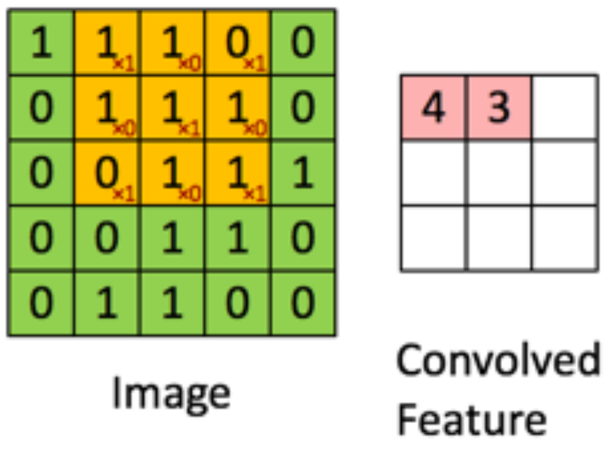
1D example:

A graphical example of 2d convolution from:

http://deeplearning.stanford.edu/wiki/index.php/Feature_extraction_using_convolution

A 3x3 block [1 0 1; 0 1 0; 1 0 1] convolved on a 5x5 image.





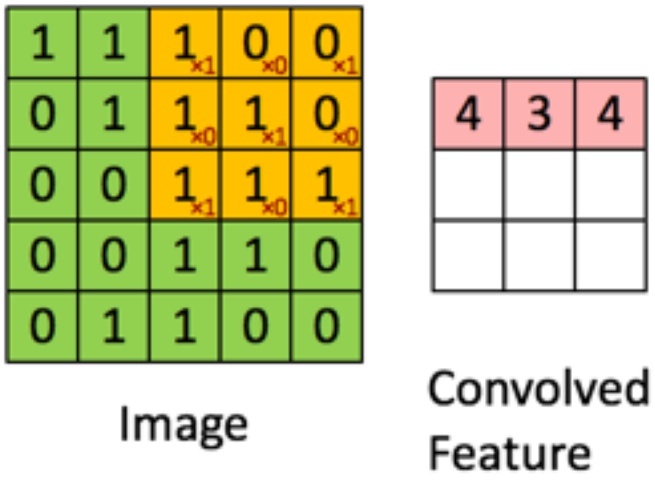


Image
Convolved Feature

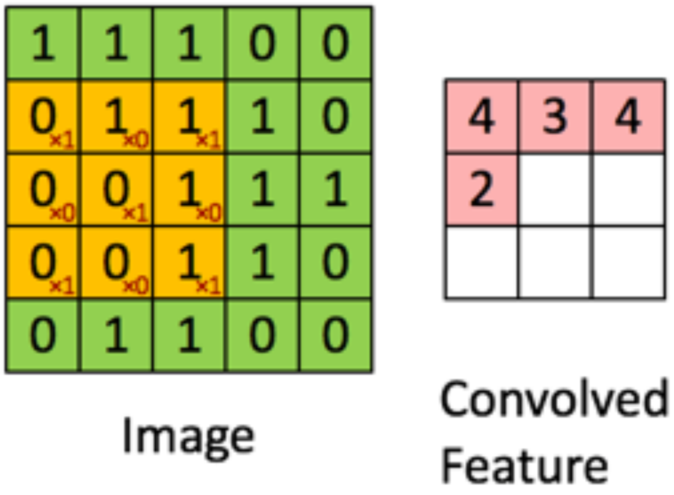


Image
Convolved Feature

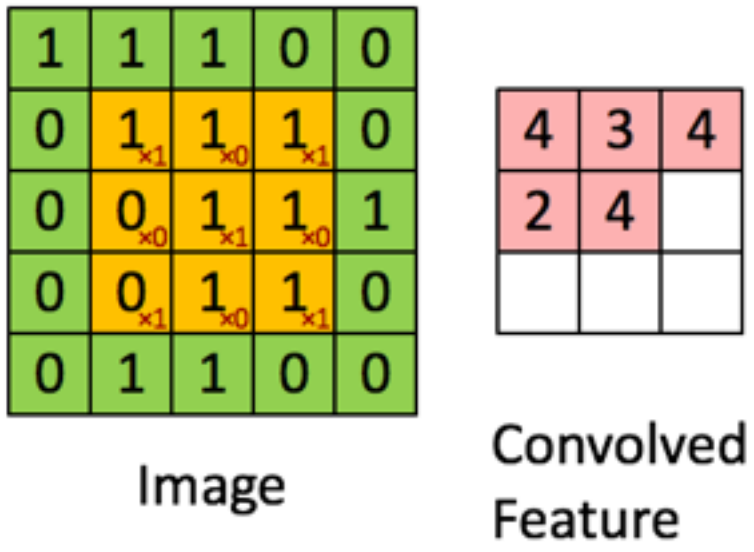


Image
Convolved Feature

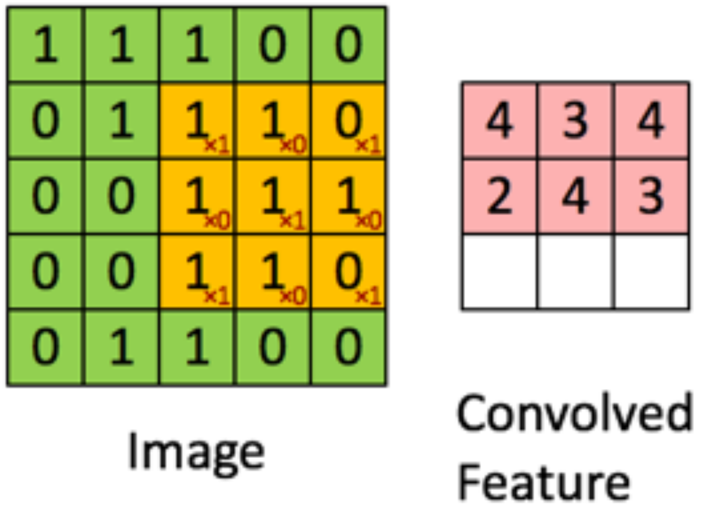


Image
Convolved Feature

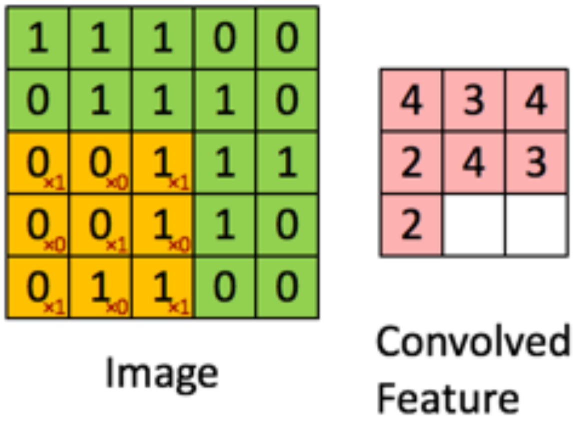


Image
Convolved Feature

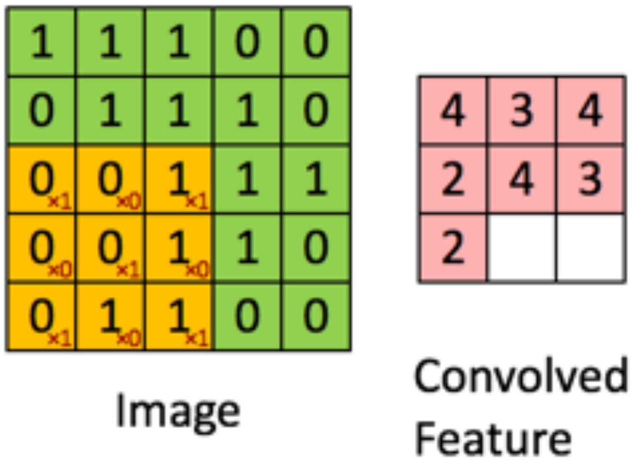


Image
Convolved Feature

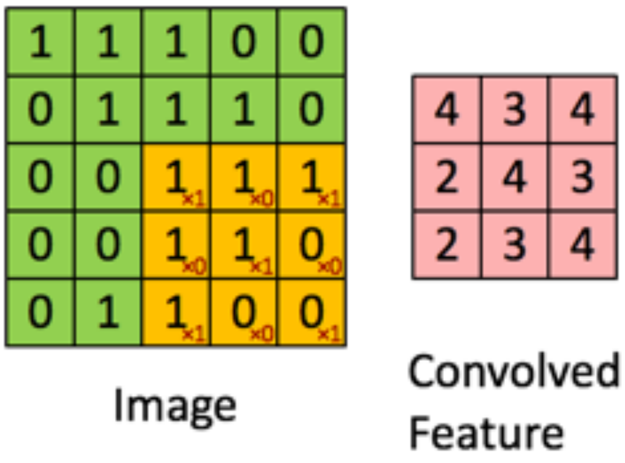
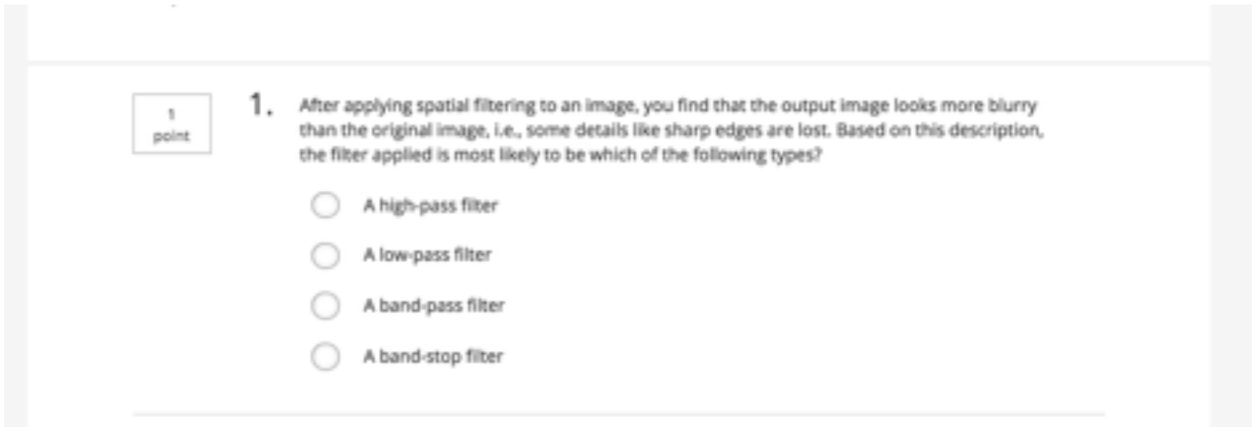


Image
Convolved Feature



1 point

1. After applying spatial filtering to an image, you find that the output image looks more blurry than the original image, i.e., some details like sharp edges are lost. Based on this description, the filter applied is most likely to be which of the following types?

- A high-pass filter
- A low-pass filter
- A band-pass filter
- A band-stop filter

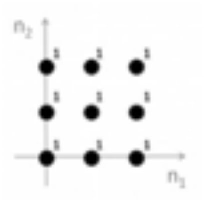
a) HPF filters out edges is wrong.
LPF filters out edges/blurry

[-1 -1 -1; -1 9 -1 ;-1 -1 -1] is the HPF. Correct

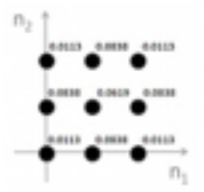
1 point

2. Which one of the following impulse responses acts a high-pass filter?

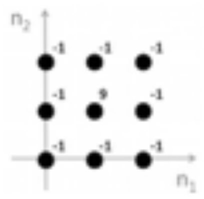
○



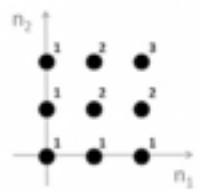
○



○



○



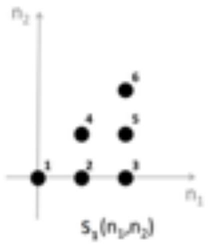
Question 3)

```
x=[1 2 3; 0 4 5; 0 0 6]
y=[1 1; 1 1]


x =

     1     2     3
     0     4     5
     0     0     6


y =

     1     1
     1     1


conv_x_y =

     1     3     5     3
     1     7    14     8
     0     4    15    11
     0     0     6     6
```

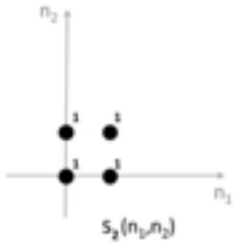
Pick answer with above. This is correct

1 point

3. What is the linear convolution of $s_1(n_1,n_2)$ and $s_2(n_1,n_2)$?

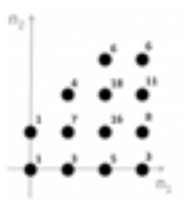


$s_1(n_1,n_2)$

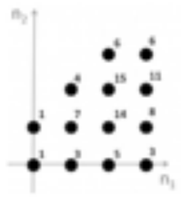


$s_2(n_1,n_2)$

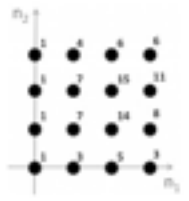
○



○

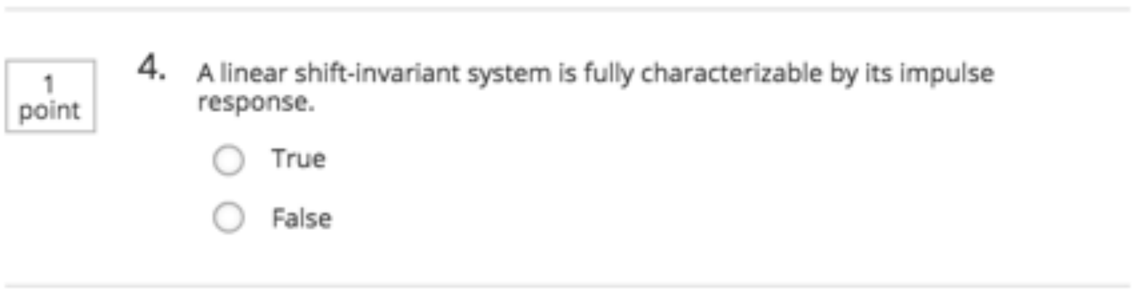


○



○ None of the above.

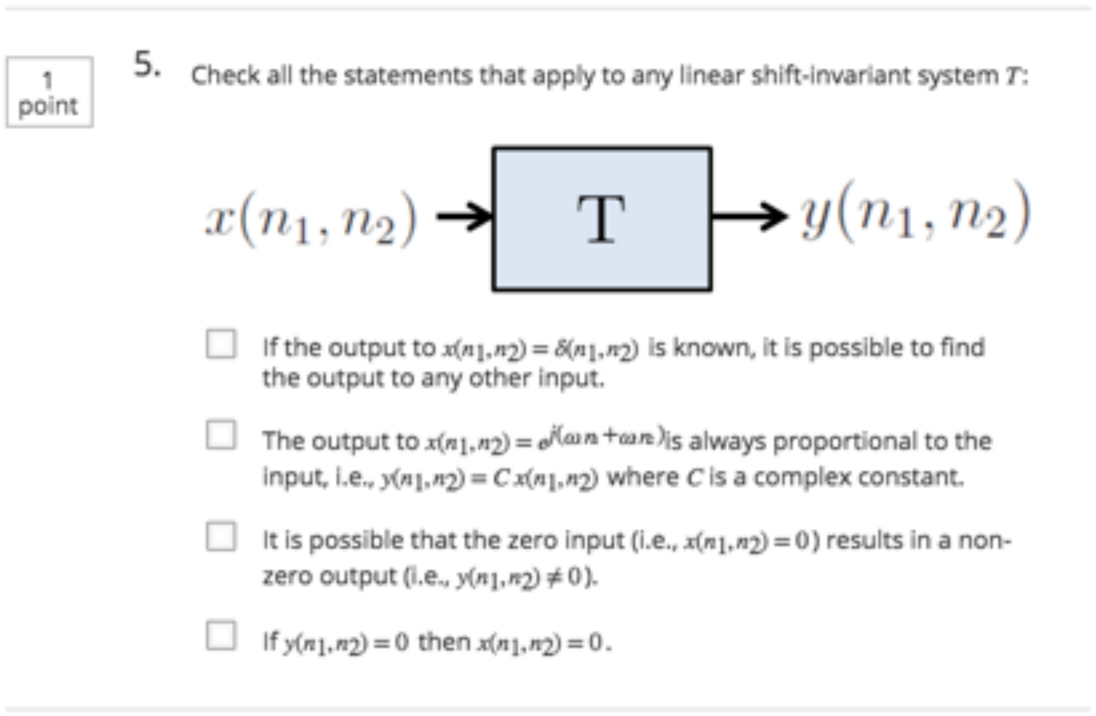
4) True; correct



1 point

4. A linear shift-invariant system is fully characterizable by its impulse response.

- ○ True
- ○ False



1 point

5. Check all the statements that apply to any linear shift-invariant system T:

$$x(n_1, n_2) \rightarrow \boxed{\mathrm{T}} \rightarrow y(n_1, n_2)$$

- ☐ If the output to $x(n_1,n_2)=\delta(n_1,n_2)$ is known, it is possible to find the output to any other input.
- ☐ The output to $x(n_1,n_2)=e^{j(\omega_1 n_1+\omega_2 n_2)}$ is always proportional to the input, i.e., $y(n_1,n_2)=C\,x(n_1,n_2)$ where C is a complex constant.
- ☐ It is possible that the zero input (i.e., $x(n_1,n_2)=0$) results in a non-zero output (i.e., $y(n_1,n_2)\neq 0$).
- ☐ If $y(n_1,n_2)=0$ then $x(n_1,n_2)=0$.

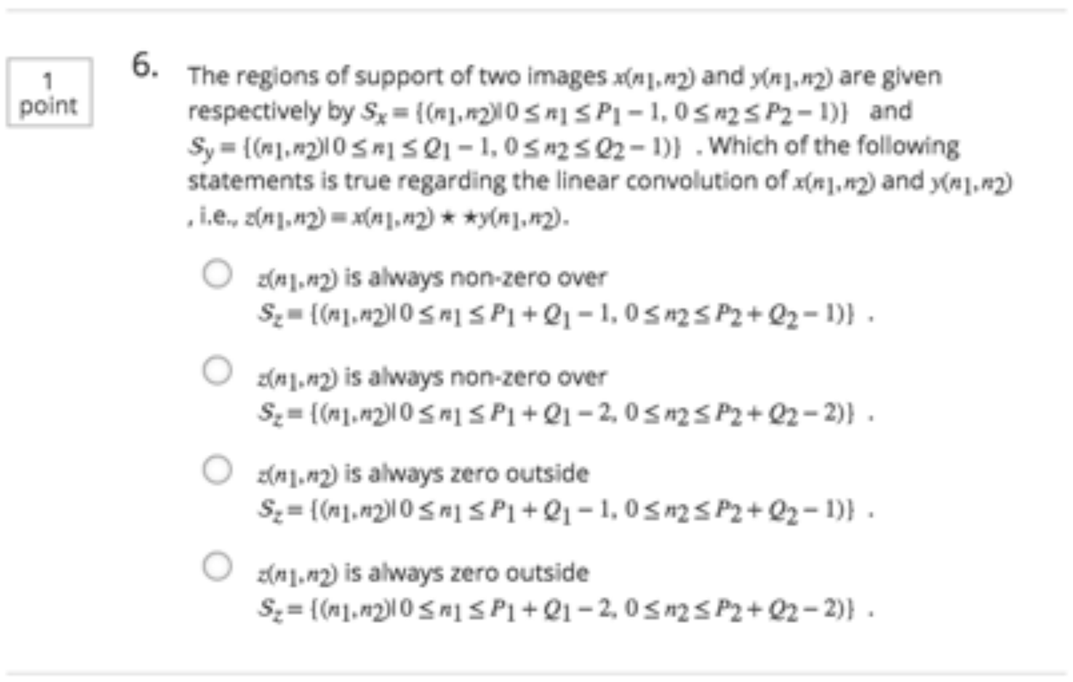
Group below is correct:

True : If the output x(n1,n2)=delta(n1,n2) is known, it is possible to find the output to any other input. This is true if the output follows the linear rules of superposition and scalar multiple.

False : The output to x(n1,n2) = exp (j(wn1+wn2)) is always porportional to the input, y(n1,n2) = Cx(n1,n2) where C is a complex constant.

False: the zero input should always be zero output. x**h if x is 0 the output is always 0.

False; if y(n1,n2)=0 then x(n1,n2)=0. y= x**h can be defined so h=null function and any input makes y 0.



1 point

6. The regions of support of two images $x(n_1,n_2)$ and $y(n_1,n_2)$ are given respectively by $S_x = \{(n_1,n_2) | 0 \le n_1 \le P_1 - 1,\ 0 \le n_2 \le P_2 - 1)\}$ and $S_y = \{(n_1,n_2) | 0 \le n_1 \le Q_1 - 1,\ 0 \le n_2 \le Q_2 - 1)\}$. Which of the following statements is true regarding the linear convolution of $x(n_1,n_2)$ and $y(n_1,n_2)$, i.e., $z(n_1,n_2) = x(n_1,n_2) \star \star y(n_1,n_2)$.

- $z(n_1,n_2)$ is always non-zero over $S_z = \{(n_1,n_2) | 0 \le n_1 \le P_1 + Q_1 - 1,\ 0 \le n_2 \le P_2 + Q_2 - 1)\}$.
- $z(n_1,n_2)$ is always non-zero over $S_z = \{(n_1,n_2) | 0 \le n_1 \le P_1 + Q_1 - 2,\ 0 \le n_2 \le P_2 + Q_2 - 2)\}$.
- $z(n_1,n_2)$ is always zero outside $S_z = \{(n_1,n_2) | 0 \le n_1 \le P_1 + Q_1 - 1,\ 0 \le n_2 \le P_2 + Q_2 - 1)\}$.
- $z(n_1,n_2)$ is always zero outside $S_z = \{(n_1,n_2) | 0 \le n_1 \le P_1 + Q_1 - 2,\ 0 \le n_2 \le P_2 + Q_2 - 2)\}$.

Zero outside P1+Q1-1; P2+Q2-1 is incorrect
nonzero over P1+Q1-1; P2+Q2-1 is incorrect

Incorrect, out of tries
5/8

1 point

7. In this problem and the next, you will implement spatial-domain low-pass filtering using MATLAB, and evaluate the difference between the filtered image and the original image using two quantitative metrics called Mean Squared Error (MSE) and Peak Signal-to-Noise Ratio (PSNR). Given two $N_1 \times N_2$ images $x(n_1,n_2)$ and $y(n_1,n_2)$, the MSE is computed as $MSE = \frac{1}{NN} \sum_{n_1=1}^{N} \sum_{n_2=1}^{N} [x(n_1,n_2) - y(n_1,n_2)]^2$.

The PSNR is defined as $PSNR = 10\log_{10}\left(\frac{MAX_I^2}{MSE}\right)$, where MAX_I is the maximum possible pixel value of the images. For the 8-bit gray-scale images considered in this problem, $MAX_I = 255$.

Follow the instructions below to finish this problem.

(1) Download the original image from here. The original image is a 256×256 8-bit gray-scale image.

(2) Convert the original image from type 'uint8' (8-bit integer) to 'double' (real number).

(3) Create a 3×3 low-pass filter with all coefficients equal to 1/9, i.e., create a 3×3 MATLAB array with all elements equal to 1/9.

(4) Low-pass filter the original image (converted to type 'double') with the filter created in step (3). This can be done using the built-in MATLAB function "imfilter". The function "imfilter" takes three arguments and returns one output. The first argument is the original image (converted to type 'double'); the second argument is the low-pass filter created in step (3); and the third argument is a string specifying the boundary filtering option. For this problem, use 'replicate' (including the single quotes) for the third argument. The output of the function "imfilter" is the filtered image.

(5) Compute the PSNR value between the original image (converted to type 'double') and the filtered image by using the formulae given above.

Enter the PSNR value (up to two decimal points).

Enter answer here

1
point

8. Repeat steps (3) through (5) in the previous question, this time using a 5×5 low-pass filter with all coefficients equal to 1/25. Enter the PSNR value (up to two decimal points).

Enter answer here