Lecture 14: Convolution and Frequency Domain Filtering

Harvey Rhody
Chester F. Carlson Center for Imaging Science
Rochester Institute of Technology
rhody@cis.rit.edu

October 27, 2005

Abstract
The impulse response of a filter describes its spatial processing behavior. This is closely tied to the global processing characteristics that are obtained using Fourier transforms. Here we focus on the relationship between the spatial and frequency domains and provide examples of alternative implementations of filters with various desirable characteristics.
Filtering by Convolution

We will first examine the relationship of convolution and filtering by frequency-domain multiplication with 1D sequences.

Let \( f(n), \ 0 \leq n \leq L - 1 \) be a data record.

Let \( h(n), \ 0 \leq n \leq K - 1 \) be the impulse response of a discrete filter.

If the sequence \( f(n) \) is passed through the discrete filter then the output is

\[
g(m) = \sum_{k=k_1}^{k_2} h(k)f(m - k), \quad -\infty < m < \infty
\]

where \( k_1 \) and \( k_2 \) are appropriate limits on the index.
Index Values

The values of \( k_1 \) and \( k_2 \) depend upon \( m \). Getting this dependence right is the detail that causes the most trouble in the implementation of convolution.

<table>
<thead>
<tr>
<th>Condition</th>
<th>Restriction</th>
</tr>
</thead>
<tbody>
<tr>
<td>( h(k) = 0 ) for ( k &lt; 0 )</td>
<td>( k_1 \geq 0 )</td>
</tr>
<tr>
<td>( f(m - k) = 0 ) for ( m - k \geq L )</td>
<td>( k_1 \geq m - L + 1 )</td>
</tr>
<tr>
<td>( h(k) = 0 ) for ( k \geq K )</td>
<td>( k_2 \leq K - 1 )</td>
</tr>
<tr>
<td>( f(m - k) = 0 ) for ( m - k &lt; 0 )</td>
<td>( k_2 \leq m )</td>
</tr>
</tbody>
</table>

These restrictions can be combined with the requirement that \( k_1 \leq k_2 \) to yield

\[
m \geq k_2 \geq k_1 \geq 0 \Rightarrow m \geq 0 \tag{1}
\]

\[
K - 1 \geq k_2 \geq k_1 \geq m - L + 1 \Rightarrow m \leq K + L - 2 \tag{2}
\]

Hence, \( g(m) = 0 \) outside the interval \( 0 \leq m \leq K + L - 2 \).
Convolution Sum

\[ g(m) = \sum_{k_1(m)}^{k_2(m)} h(k) f(m - k), \quad 0 \leq m \leq K + L - 2 \]

= 0, elsewhere

where the limits are described by the following tables.

| Case $K < L$          | Range of \( m \) | \( k_1 \) | \( k_2 \) |
|-----------------------|------------------|---------|
|                       | $0 \leq m \leq K - 1$ | 0       | \( m \) |
|                       | $K \leq m \leq L - 1$ | 0       | \( K - 1 \) |
|                       | $L \leq m \leq K + L - 2$ | \( m - L + 1 \) | \( K - 1 \) |

| Case $K \geq L$       | Range of \( m \) | \( k_1 \) | \( k_2 \) |
|-----------------------|------------------|---------|
|                       | $0 \leq m \leq L - 1$ | 0       | \( m \) |
|                       | $L \leq m \leq K - 1$ | \( m - L + 1 \) | \( m \) |
|                       | $K \leq m \leq K + L - 2$ | \( m - L + 1 \) | \( K - 1 \) |
Periodic Convolution

The complexity of the convolution calculation can be reduced by using periodic functions.

Let $f(n)$ and $h(n)$ be padded with zeros to length $P \geq L + K - 1$ and then repeated with period $P$. The resulting periodic functions will be called $f_P(n)$ and $h_P(n)$.

The periodic convolution is defined as

$$g_P(m) = \sum_{k=0}^{P-1} h_P(k) f_P(m - k)$$

The result is periodic with period $P$.

$$g_P(m + jP) = \sum_{k=0}^{P-1} h_P(k) f_P(m + jP - k) = \sum_{k=0}^{P-1} h_P(k) f_P(m - k)$$
Shown at the right are two functions that we would like to process by convolution. 

\( f(n) \) has length \( L = 6 \) and \( h(n) \) has length \( K = 5 \).

The functions have been extended by zero-padding to length \( P = L + K - 1 = 10 \).
Example

The convolution of \( f_P(n) \) with \( h_P(n) \) is shown at the right. One period of \( f_P(n) \) is shown at the top of each column. The cyclic shifts \( h_P(n - m) \) of the filter response are shown in rows 2-6.

The value of \( g_P(m) \) is found by multiplying \( h_P(m - n) \) with \( f_P(n) \) and summing. The result for each shift is listed beside the corresponding filter shift.
Periodic Convolution

It is readily demonstrated that

\[ g_P(m) = g(m), \quad 0 \leq m \leq P - 1 \]

Thus, we can use circular convolution to compute the finite convolution.

The periodic functions \( f_P(n) \) and \( h_P(n) \) have the discrete Fourier transforms

\[
F(u) = \frac{1}{P} \sum_{n=0}^{P-1} f_P(n)e^{-i2\pi un/P} \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad

Note the dependence on \( P \), rather than on \( L \) or \( K \). Note also that \( F(u) \) and \( H(u) \) are periodic with period \( P \).
The functions involved in the periodic convolution of $f$ and $h$ are shown above. The period $P = 10$ is the minimum length for these functions. The result is a periodic function that is in agreement with $g$ in the interval $0 \leq n \leq K + L - 1$. 
The DFT of $g_P(m)$ can be computed by

$$G(u) = \frac{1}{P} \sum_{m=0}^{P-1} g_P(m) e^{-i2\pi um/P}$$

$$= \frac{1}{P} \sum_{k=0}^{P-1} \sum_{m=0}^{P-1} h_P(k) f_P(m - k) e^{-i2\pi um/P}$$

Make the change of index $m = k + n$ and rearrange.

$$G(u) = \frac{1}{P} \sum_{k=0}^{P-1} h_P(k) \sum_{n=-k}^{P-k-1} f_P(n) e^{-i2\pi u(n+k)/P}$$

$$= \left( \frac{1}{P} \sum_{k=0}^{P-1} h_P(k) e^{-i2\pi uk/P} \right) \left( \sum_{n=-k}^{P-k-1} f_P(n) e^{-i2\pi un/P} \right)$$

The terms $n = -k, \ldots , -1$ in the last sum can be replaced by $n = P - k, \ldots , P - 1$ because the summand is periodic. Hence . . .
The sum then simplifies to the product

\[ G(u) = P \left( \frac{1}{P} \sum_{k=0}^{P-1} h_P(k) e^{-i2\pi uk/P} \right) \left( \frac{1}{P} \sum_{n=0}^{P-1} f_P(n) e^{-i2\pi un/P} \right) \]

\[ = PH(u)F(u) \]

1. By construction, \( f_P(n) \), \( h_P(n) \) and \( g_P(n) \) are all periodic with period \( P \).
2. It is required that \( P \geq L + K - 1 \).
3. \( F(u) \), \( H(u) \) and \( G(u) \) are all complex periodic functions with period \( P \).
4. \( F(u) \) can be computed by padding \( f(n) \) with zeros to length \( P \) and then doing a DFT.
5. A similar statement is true for \( H(u) \) and \( G(u) \).
6. \( u \) takes on integer values.
The transforms $F(u)$, $H(u)$ and $G(u)$ can be displayed in the complex plane. The points are numbered with the value of the index $u$, $0 \leq u \leq P - 1$. Because the functions are periodic, they follow the same cycle repetitively.
Power Relationships

The mean-squared value of $f(n)$ over the interval of length $P$ is

$$\frac{1}{P} \sum_{n=0}^{P-1} f_P(n)f_P^*(n) = \frac{1}{P} \sum_{n=0}^{P-1} \sum_{u=0}^{P-1} \sum_{v=0}^{P-1} F(u)F^*(v)e^{i2\pi(u-v)n/P}$$

$$= \sum_{u=0}^{P-1} \sum_{v=0}^{P-1} F(u)F^*(v) \left( \frac{1}{P} \sum_{n=0}^{P-1} e^{i2\pi(u-v)n/P} \right)$$

The term in parentheses is $\delta_{u,v}$ Hence,

$$\frac{1}{P} \sum_{n=0}^{P-1} |f_P(n)|^2 = \sum_{u=0}^{P-1} |F(u)|^2$$

1. Because $f(n) = f_P(n)$ in $0 \leq n \leq P - 1$, the above statement is also valid for $f(n)$.
2. An equivalent statement is true for $h_P(n)$ and $g_P(n)$. 
Example

The power calculations for the waveforms and their transforms are shown in the table below. In each case, the same result is obtained in both domains.

|                               | $\frac{1}{P} \sum_{n=0}^{P-1} |f_P(n)|^2 = 5.5$ | $\frac{1}{P} \sum_{n=0}^{P-1} |h_P(n)|^2 = 0.7$ | $\frac{1}{P} \sum_{n=0}^{P-1} |g_P(n)|^2 = 63.9$ |
|-------------------------------|-------------------------------------------------|-------------------------------------------------|-------------------------------------------------|
|                               | $\sum_{u=0}^{P-1} |F(u)|^2 = 5.5$ | $\sum_{u=0}^{P-1} |H(u)|^2 = 0.7$ | $\sum_{u=0}^{P-1} |G(u)|^2 = 63.9$ |
Filtering of 2D Functions

Let $f(x, y)$ be a $L_1 \times L_2$ array that represents the brightness values of an image.

Let $h(x, y)$ be a $K_1 \times K_2$ array that represents the impulse response of a spatial filter.

The result of filtering $f$ with $h$ is the array $g(x, y)$. By analogy with the 1D example, we find that $g(x, y)$ has size $(L_1 + K_1 - 1, L_2 + K_2 - 1)$.

The convolution operation can be most conveniently written in terms of functions that are periodic with period $(P_1, P_2)$, where

\[
\begin{align*}
P_1 &\geq L_1 + K_1 - 1 \\
P_2 &\geq L_2 + K_2 - 1
\end{align*}
\]
**Periodic Convolution**

The periodic convolution is defined as

\[
g_P(x, y) = \sum_{j=0}^{P_1-1} \sum_{k=0}^{P_2-1} h_P(j, k) f_P(x - j, y - k)\]

The result is periodic with period \(P_1, P_2\).

\[
g_P(x + aP_1, y + bP_2) = \sum_{j=0}^{P_1-1} \sum_{k=0}^{P_2-1} h_P(j, k) f_P(x + aP_1 - j, y + bP_2 - k)\]

\[
= \sum_{j=0}^{P_1-1} \sum_{k=0}^{P_2-1} h_P(j, k) f_P(x - j, y - k)\]

\[
= g_P(x, y)\]
Periodic Convolution

1. Assume that \( f(x, y) \) is given only for \( 0 \leq x \leq L_1 - 1 \) and \( 0 \leq y \leq L_2 - 1 \).
2. Assume that \( h(x, y) \) is given only for \( 0 \leq x \leq K_1 - 1 \) and \( 0 \leq y \leq K_2 - 1 \).
3. Without loss of generality, we can assume any extension we wish outside the region of definition.
4. The convolution \( g = f \star h \) has size \((L_1 + K_1 - 1, L_2 + K_2 - 1)\).
5. We can construct periodic functions \( f_P(x, y) \) and \( h_P(x, y) \) that are defined on the primary frame \((P_1 \geq L_1 + K_1 - 1, P_2 \geq L_2 + K_2 - 1)\) by padding with zeros. In the examples we will use \( P_1 = L_1 + K_1 - 1, \ P_2 = L_2 + K_2 - 1 \)
6. Periodic convolution of \( f_P(x, y) \) with \( h_P(x, y) \) produces a periodic function \( g_P(x, y) \) that is equal to \( g(x, y) \) for \( 0 \leq x \leq P_1 - 1, \ 0 \leq y \leq P_2 - 1 \).
Periodic Convolution

7. Periodic convolution is equivalent to multiplication of the transforms of the periodic spatial functions.

8. The discrete Fourier transforms are periodic with period \((P_1, P_2)\).

\[
F(u, v) = \frac{1}{P_1 P_2} \sum_{x=0}^{P_1-1} \sum_{y=0}^{P_2-1} f_P(x, y) e^{-i2\pi \left( \frac{xu}{P_1} + \frac{vy}{P_2} \right)}
\]

\[
= \frac{1}{P_1 P_2} \sum_{x=0}^{L_1-1} \sum_{y=0}^{L_2-1} f(x, y) e^{-i2\pi \left( \frac{xu}{P_1} + \frac{vy}{P_2} \right)}
\]

Similar expressions hold for the other variables.
Algorithm

1. Read the image array $f(x, y)$ and determine the dimensions $(L_1, L_2)$.
2. Define the filter $h(x, y)$ with dimensions $(K_1, K_2)$.
3. Construct zero-padded arrays $f_P(x, y)$ and $h_P(x, y)$ with dimensions $(P_1, P_2)$.
4. Compute $F(u, v) = \text{FFT}(f_P(x, y))$ and $H(u, v) = \text{FFT}(h_P(x, y))$.
5. Compute $G(u, v) = P_1 P_2 F(u, v) H(u, v)$
6. Compute $g_P(x, y) = \text{FFT}^{-1}(G(u, v))$. 
Algorithm Example

N=20 & M=20
x=(findgen(N)-N/2)#Replicate(1,M)
y=(findgen(M)-M/2)#Replicate(1,N)
fxy=exp(-(x^2/15+y^2/5))

:STEP 2: DEFINE THE FILTER ARRAY WITH SIZE 3X3.
hxy=[[-1,0,-1],[0,4,0],[-1,0,-1]]

:STEP 3: PAD THE ARRAYS TO (22,22).
hp=zeropad(hxy,22,22)
fp=zeropad(fxy,22,22)

:STEP 4: DO THE FFT's
ff=FFT(fp)
hf=fft(hp)

:STEP 5: COMPUTE G(u,v)
gf=hf*ff*22*22

:STEP 6: COMPUTE gp
gp=float(fft(gf,/inverse))
Example

Function $f(x, y)$.

Function $h_P(x, y)$.

Function $g_P(x, y)$.

Function $|F(u, v)|$.

Function $|H(u, v)|$.

Function $|G(u, v)|$. 
Function zeroPad, A, N, M, Center=center

; B = zeroPad(A, N, M) returns an array B of size NxM such that B[i,j] = A[i,j]
; where A is defined and B[i,j] = 0 where A[i,j] is not defined. B is the same
; type as A.
;
; If A is a scalar or a row vector then zeroPad can be called with a single
; dimension argument. B = zeroPad(A, N) returns a row vector of length N.
;
; B = zeropad(A) returns an array that has a single ring of zeros around A.
;
; It is an error if N or M is smaller than the corresponding dimension of A.
;
; KEYWORDS
;
; CENTER: Set the keyword center if the array should be centered in the
; field of zeros.
;
; HISTORY
; Written by H. Rhody September, 1999
; Modified to include additional error checks and CENTER keyword
; Modified to handle B = zeropad(A) case. 9.23.2005 HR
;-
Zeropad Examples

IDL> print, zeropad(4)
    0  0  0
    0  4  0
    0  0  0

IDL> print, zeropad([1,2,3])
    0  0  0  0  0
    0  1  2  3  0
    0  0  0  0  0

IDL> print, zeropad([1,2,3],7)
    1  2  3  0  0  0  0

IDL> print, zeropad([1,2,3],7,/center)
    0  0  1  2  3  0  0

IDL> print, zeropad(indgen(3,4),7,8,/center)
    0  0  0  0  0  0  0
    0  0  0  0  0  0  0
    0  0  0  1  2  0  0
    0  0  3  4  5  0  0
    0  0  6  7  8  0  0
    0  0  9  10 11 0  0
    0  0  0  0  0  0  0
    0  0  0  0  0  0  0
Comparison with Convolution

1. Spatial filtering can be carried out by convolution.
2. IDL contains a fast convolution routine, CONVOL.
3. Care must be taken to do mathematical convolution with CONVOL.

Steps to do convolution of arrays $A$ and $B$ with CONVOL.

1. Determine the dimensions $(L_1, L_2)$ and $(K_1, K_2)$ of $A$ and $B$.
2. Construct $A_p$ by zero-padding $A$ to size $(L_1 + K_1 - 1, L_2 + K_2 - 1)$.
   
   \[ \text{Ap = zeropad}(A, L_1 + K_1 - 1, L_2 + K_2 - 1) \]
3. Compute $CP =$ CONVOL($A_p, B, \text{CENTER}=0,/\text{EDGE}_\text{ZERO}$).

This procedure replaces a more complicated method that was required before the \text{EDGE}_\text{ZERO} keyword was introduced in IDL 6.2.

The “old” method is used in the \text{CONVOLVE} program below, which makes it compatible with older versions of IDL.
CONVOLVE Program

FUNCTION CONVOLVE,A,M,SCALE,_EXTRA=extra

; B=CONVOLVE(A,M,S) returns an array B that is the 2D 
; convolution of the arrays A and M. The convolution calculation
; is actually carried out by the IDL CONVOL routine.
;
; An array AP is constructed by padding A, then
; C=CONVOL(AP,M,S,CENTER=0,/EDGE_TRUNCATE) is computed
; and then C is trimmed to provide the proper output.
;
; The size of the returned array is equal to
; Ac+Mc-1 by Ar+Mr-1, where Ac x Ar is the size of
; A and Mc x Mr is the size of M.
;
; If S is provided then the results are scaled by
; passing S to the CONVOL routine. Otherwise, the
; scale value S=1 is used.
;
; CONVOL keywords are passed via _EXTRA.
;
; H. Rhody
; September, 2001
; -
CONVOLVE (cont)

IF N_PARAMS() LT 2 THEN MESSAGE,'Incorrect number of parameters.'
SA=Size(A)
SM=Size(M)
typeA=SA[SA[0]+1]
typeM=SM[SM[0]+1]
IF N_PARAMS() LT 3 THEN SCALE=1

; Determine the width and height parameters. The case of a 1D array
; has to be handled differently.
CASE SA[0] OF
1: BEGIN
   WA=SA[1] & HA=1
END

2: BEGIN
END
ENDCASE

CASE SM[0] OF
1: BEGIN
   WM=SM[1] & HM=1
END

2: BEGIN
END
ENDCASE
CONVOLVE (cont)

;Construct a zero-padded array to use in the convolution operation
SAP=[2,WA+2*(WM-1),HA+2*(HM-1),MAX([typeA,typeM]),(WA+2*(WM-1))*(HA+2*(HM-1))]
AP=Make_Array(Value=0,Size=SAP)
AP[WM-1:WM+WA-2,HM-1:HM+HA-2]=A

;Do the convolution of AP with M
AC=CONVOL(AP,M,SCALE,CENTER=0,/EDGE_TRUNCATE,_EXTRA=extra)

;Trim and return the convol result.
CSTART=WM-1
RSTART=HM-1
RETURN,AC[CSTART:CSTART+WM+WA-2,RSTART:RSTART+HM+HA-2]
END
Comparison of Transform and Convolution Methods

Shown above is the output array and the magnitude of the difference between the transform and convolution methods. All of the differences are less than $3 \times 10^{-7}$. 
Power Calculations

The mean-squared value can be computed in either the spatial or frequency domain.

\[
\frac{1}{P_1P_2} \sum_{x=0}^{P_1-1} \sum_{y=0}^{P_2-1} |f_P(x, y)|^2 = \sum_{u=0}^{P_1-1} \sum_{v=0}^{P_2-1} |F(u, v)|^2
\]

For the preceding example we have

\[
\frac{1}{22^2} \sum_{x=0}^{21} \sum_{y=0}^{21} |f_P(x, y)|^2 = 0.0281
\]

\[
\sum_{u=0}^{21} \sum_{v=0}^{21} |F(u, v)|^2 = 0.0281
\]
Power Calculations (cont)

The output image has the frequency-domain relationship

\[ G(u, v) = P_1 P_2 F(u, v) H(u, v) \]

Hence, the power can be computed by

\[
\frac{1}{P_1 P_2} \sum_{x=0}^{P_1-1} \sum_{y=0}^{P_2-1} \left| g(x, y) \right|^2 = \sum_{u=0}^{P_1-1} \sum_{v=0}^{P_2-1} \left| G(u, v) \right|^2
\]

\[ = (P_1 P_2)^2 \sum_{u=0}^{P_1-1} \sum_{v=0}^{P_2-1} \left| F(u, v) \right|^2 \left| H(u, v) \right|^2 \]

\[ = .0147 \quad \text{for the example above} \]
Image Filtering Example

Image arrays can be processed by either spatial or transform filtering. Just substitute the image array for $f(x, y)$ in the above techniques.

You may need to rescale the amplitude of the final array to fit the range [0,255] for display.