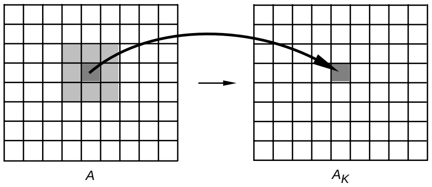
MATH 261, Spring 2001 **Due Date:**____________ **Name(s):** ________________________

Honors Project 6: Convolution and Digital Image Processing

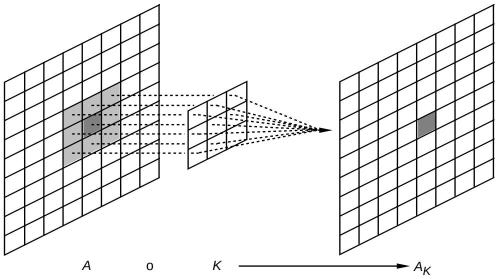
Narrative

If you have not already done so, do Honors Project 5.

The convolution of two arrays of real numbers is another array of real numbers. We can describe the process of convolution conceptually, geometrically, and analytically. Conceptually (see the figure below), the convolution of an $N \times M$ array A with a 3×3 array (or *kernel*) K is the array A_K whose (row,column)-entry $A_K(R, C)$ is a weighted combination of the (row,column)-entry $A(R, C)$ of A, together with all its immediate neighbors; the weights are the entries of the kernel, and $A_K(R, C)$ is the sum of products of the entries of A with the corresponding entries of K.



Geometrically, A_K is created by centering K over each entry $A(R, C)$ of A, and then "filtering" $A(R, C)$, as well as the value of A at each immediate neighbor of (R, C), through K; "filtering" is accomplished by summing products of the entries of A with the corresponding entries of K.



Computationally, for each R and C:

$$A_K(R, C) = \sum_{i=-1}^{1} \sum_{j=-1}^{1} K(2+i, 2+j) * A(R+i, C+j)$$

$$\begin{aligned}
&= K(1,1) * A(R-1, C-1) + K(1,2) * A(R-1, C) + K(1,3) * A(R-1, C+1) \\
&\quad + K(2,1) * A(R-1, C-1) + K(2,2) * A(R-1, C) + K(2,3) * A(R-1, C+1) \\
&\quad + K(3,1) * A(R+1, C-1) + K(3,2) * A(R+1, C) + K(3,3) * A(R+1, C+1).
\end{aligned}$$

(For completeness, we assume N and M are both larger than 3, and that we take $A(R, C)$ to be 0 in this sum if $R < 0$ or $R > N$, or if $C < 0$ or $C > M$.)

As long as K is the same type as A, A_K will be of the same type as A. If, however, K is *not* the same type as A, then some small modification might be necessary so that A_K *will be* of the same type as A. For example, if A is an integer array, but K is not, then we might have to round the entries of A_K, or take their floor or ceiling, to obtain an array whose entries are again integers.

Tasks

1. a) Type the following command lines into Maple in the order in which they are listed.

```
> # Honors Project 6:  Convolution and Digital Image Processing
> restart:  with(linalg):
> A := matrix(12,12,(Row,Col)->0);
> for Row from 3 to 10 do for Col from 3 to 10 do A[Row,Col] := 9 od; od;
> A := evalm(A);
> K := matrix(3,3,(Row,Col)->1/9);
> A_K := matrix(12,12,(Row,Col)->0):
> for Row from 2 to 11 do for Col from 2 to 11 do
    Tally := 0:
    for i from -1 to 1 do for j from -1 to 1 do
      Tally := Tally+K[2+i,2+j]*A[Row+i,Col+j]:  od:  od:
    A_K[Row,Col] := floor(Tally):
  od:  od:
> A_K := evalm(A_K);
```

   In this code we declare a matrix A and fill the entries of `A` with integers between 0 and 9. If we think of A as a digital image in which a 0 represents white, a 9 represents black, and numbers between 0 and 9 represent shades of gray (increasing in intensity from 1 to 8) then A is an image of a black square surrounded by a white border. We then define a kernel K all of whose entries are 1/9, and compute the convolution A_K. (The function $floor$ associates to each real number x the largest real number smaller than x.)

   b) Describe in words the effect of convolving A with K.

2. Repeat both parts of Task 1 using the following kernels:

   a) $K = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$ b) $K = \begin{pmatrix} 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 \end{pmatrix}$ c) $K = \begin{pmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{pmatrix}$ d) $K = \frac{1}{16}\begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix}$

   Feel free to experiment with different images A if that helps describe the effect of convolving A with K.

Comments

1. The convolution of two arrays is a discrete form of continuous convolution: if $f = f(x,y)$ and $K = K(x,y)$ are continuous functions of 2 real variables, for example, then the convolution $f_K = f_K(x,y)$ of f with K is defined by

$$f_K = f_K(x,y) = \int\int K(u,v)f(x-u,y-v)\ du\ dv.$$

2. There are numerous identities involving convolutions (of both arrays and functions) as long as no rounding or other such operation is invoked. For example, for arrays:

$$(A+B)_K = (A+B)_K, \quad \text{and} \quad (kA)_K = kA_K$$

$$A_{K+L} = A_K + A_L, \quad \text{and} \quad A_{kK} = kA_K$$

3. In digital image processing, arrays such as A are generally much larger than 12×12, and each entry of A is generally an integer in the range of 0..255. Further, the convolution A_K can be defined if dim K is an odd integer greater than 3; the larger dim K, the wider the scope of K in the convolution. The sum of the entries of K is often chosen to be between 0 and 1: this is to avoid introducing any "energy" into the convolution (which will produce entries in A_K that are "too often" above 255, or "too often" below 0).

4. Convolutions on digital images are important since they represent operations that are more general than the operations that can be performed on analog images. The operations that can be performed on analog images are pointwise operations; these operations include changing brightness and contrast. Digital images can be modified (through convolution) by neighborhood operations; these operations go beyond pointwise operations, and include smoothing, sharpening, and edge detection. One of the reasons for the current popularity of digital TV is that much more can be done to improve the image quality of a digital image using convolution, than can be done to improve the image quality of an analog image.

5. There are numerous software tools available that are particularly well adapted for further study of image processing. These include MatLab, NIH-Image, and J-Image.

Contributor: Dr. Kris A. Dines
X-Data Corporation, Indianapolis, IN