A combinatorial interpretation of Gaussian blur

Travis Dillon

Lawrence University

A combinatorial interpretation of Gaussian blur

Travis Dillon*

Lawrence University

ABSTRACT. Gaussian blur is a commonly-used method to filter image data. This paper introduces the collapsing sum, a new operator on matrices that provides a combinatorial interpretation of Gaussian blur. We study the combinatorial properties of this operator and prove the explicit relation between Gaussian blur and the collapsing sum.

1. INTRODUCTION

Image data, and data in general, is often filtered to remove *noise*, random fluctuations that hide the underlying pattern. For images, one of the most common solutions is to apply Gaussian blur, which smooths the data to remove noise.

Because of its use, there has been much interest in discovering efficient algorithms for Gaussian blur [1, 2, 3]. Waltz and Miller [3] in particular provide a clear example of the ways in which properties of binomial coefficients can be leveraged to create such an algorithm. An analysis of their algorithm in Section 2 leads to the following definitions.

Definition 1.1. Let A be a real $m \times n$ matrix. If $m \geq 2$, then the $(m-1) \times n$ matrix $\sigma_{\downarrow}(A)$ has entries

$$\sigma_{\downarrow}(A)_{i,j} = a_{i,j} + a_{i+1,j}.$$

If $n \geq 2$, then the $m \times (n-1)$ matrix $\sigma_{\rightarrow}(A)$ has entries

$$\sigma_{\rightarrow}(A)_{i,j} = a_{i,j} + a_{i,j+1}.$$

Finally, the matrix $\sigma(A) := \sigma_{\downarrow} \circ \sigma_{\rightarrow}(A) = \sigma_{\rightarrow} \circ \sigma_{\downarrow}(A)$ is the *collapsing sum* of A.

The collapsing sum captures mathematically what Waltz and Miller describe computationally in [3]. In this paper, we establish the connection between the collapsing sum and Gaussian blur and provide a theoretical study of the combinatorial properties of this operator.

Section 3 provides the main combinatorial analysis of the operator. We recast the collapsing sum (and therefore Gaussian blur) in terms of matrix multiplication and define a

Corresponding author

new class of matrices called coefficient matrices that generalize Gaussian blur. The section culminates in Theorem 3.12, which explicitly describes the connection between the collapsing sum and Gaussian blur.

In the remainder of the paper, we turn to the purely combinatorial properties of this operator; for example, we completely describe the fully-collapsed state of Toeplitz matrices (see Proposition 4.3). Finally, we discuss generalizations of Gaussian blur in connection to Waltz and Miller's algorithm.

2. Background

As the collapsing sum will be motivated by Gaussian blur, we begin with a description of image filtering. Grayscale images are stored as matrices: Shades of gray are represented as numbers in a particular range (for example, integers from 0 to 255, or real numbers from 0 to 1), and each entry represents a pixel.[1] We will consider only grayscale images, but this is not an artificial restriction; the same techniques are used to apply a filter to color images. The data for color images are stored as three separate values of red, green, and blue. Applying a filter to a color image consists of separating the data into three matrices by color type, applying the filter to each, and recombining.

It may be the case that the image contains noise, so that the pixel values are randomly perturbed by environmental factors. Because noise is random, it seems possible to eliminate it by averaging pixel values in the neighborhood of a central pixel. This process is known as filtering.

Filters are applied in a process called convolution. The matrix that represents the filter is called a *kernel matrix*. Typically, kernel matrices are square with dimensions $(2r + 1) \times (2r + 1)$. The integer r is the *radius* of the filter and controls the size of the neighborhood. For simplicity in the convolution formula, kernel matrices are indexed so that the central entry has coordinates (0,0). Convolving the kernel matrix $K = (k_{i,j})$ with an $m \times n$ image matrix A returns the $m \times n$ matrix $K * A$ with entries

$$(K * A)_{p,q} := \sum_{i,j=-r}^{r} k_{i,j} \cdot a_{p-i,q-j}.$$

The convolution can be equivalently expressed as

$$(K * A)_{p,q} = \sum_{\substack{i+k=p \\ j+\ell=q}} k_{i,j} \cdot a_{k,\ell}.$$

We require that $\sum_{i,j} k_{i,j} = 1$ so that the overall intensity of the image does not change.

As written, however, convolution is not well-defined when $a_{p,q}$ is near a boundary of A. In these cases, the convolution formula requires values of entries that don't exist, such as $a_{-1,0}$. To fix this problem, we use what are called *edge-handling techniques*. In this paper, we only consider two common techniques: extending A to have values beyond its edges or applying the filter to only those pixels for which convolution is defined (the latter is

---

[1]Whether 0 represents black or white depends on the application; in printing, 0 represents white, whereas in computing, 0 represents black. We won't need to pick between these conventions for our purposes.

called *cropping*).[2] To apply a kernel matrix of radius r to all pixels in an $m \times n$ matrix A, we need to extend A by r rows and columns on each side, to a matrix of size $(m+2r) \times (n+2r)$, where the central $m \times n$ block is the matrix A. The filter is applied to each pixel in the central $m \times n$ block of the enlarged matrix.

If extension is chosen as the edge-handling technique, let A' denote the corresponding extension of A. If cropping is chosen as the edge-handling technique, then set $A' = A$. Applying the filter to A with the chosen edge-handling technique is equivalent to applying the filter to A' with cropping.

The simplest blur filter is the *box blur*. Let $J_{m \times n}$ represent the $m \times n$ matrix with each entry equal to 1, and abbreviate $J_{n \times n}$ by J_n.

Definition 2.1. The kernel matrix B_{2r+1} for the box blur of radius r is $(2r+1)^{-2} J_{2r+1}$.

As a visual example, consider the following image.

FIGURE 1. A grayscale image.

The results of applying box blurs with radii of 1, 2, and 3, respectively, to this image are shown below.

FIGURE 2. Box blur of a grayscale image.

One problem with box blurs, especially ones of large radius, is that pixels are weighted the same regardless of their distance from the central pixel. It makes sense to weight closer pixels more heavily than distant pixels: Pixels that are closer to each other will contain more information about each other than those that are farther away. Because of this, the *Gaussian blur*, which takes this into account, is more commonly used. The values for the Gaussian blur kernel matrix are derived from the two-dimensional Gaussian curve

$$f(x,y) = \frac{1}{2\pi s^2} e^{-\frac{x^2+y^2}{2s^2}},$$

where s represents the standard deviation of the distribution. Sometimes values are directly sampled from this function, but they are often approximated using binomial coefficients.

---

[2]See https://en.wikipedia.org/wiki/Kernel_(image_processing) for a list of edge-handling techniques.

Definition 2.2. The $(2r + 1) \times (2r + 1)$ kernel matrix G_{2r+1} of the approximate Gaussian blur with radius r has entries, for each $-r \le i, j \le r$, of

$$(G_{2r+1})_{i,j} = \frac{1}{4^{2r}} \binom{2r}{i + r} \binom{2r}{j + r}. \tag{1}$$

Example 2.3. The kernel matrix for the 5×5 approximate Gaussian blur is

$$G_5 = \frac{1}{256} \begin{pmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{pmatrix}.$$

Notice that the pixels near the center are weighted highest, and that the values taper off toward the edges. Applying Gaussian blurs of radii 1, 2, and 3, respectively, to our example image from above results in the images below. The images appear smooth, while each individual element of the image remains clear.

FIGURE 3. Gaussian blur of a grayscale image.

Each Gaussian blur kernel matrix can be decomposed into the product of a row vector and a column vector. Since it is much faster to compute smaller convolutions than large ones, Gaussian blur algorithms break the computation into two smaller convolutions: one with the row vector, and one with the column vector.

In [3], Waltz and Miller develop an algorithm for computing Gaussian blur that is more efficient than simple decomposition. The key observation that the authors use is that Gaussian blurs of larger radius can be created through repeated convolution with Gaussian blurs of smaller radius. Their algorithm decomposes the Gaussian blur kernel matrix into a row vector and a column vector, and it decomposes each of these vectors into the repeated convolution of the matrices $\begin{pmatrix} 1 & 1 \end{pmatrix}$ and $\begin{pmatrix} 1 & 1 \end{pmatrix}^T$, respectively. With a bit of clever programming, Waltz and Miller created an algorithm that runs much faster than one that only uses the decomposition property.

Convolution by the matrices $\begin{pmatrix} 1 & 1 \end{pmatrix}$ and $\begin{pmatrix} 1 & 1 \end{pmatrix}^T$ corresponds to the operations $\sigma_\rightarrow$ and $\sigma_\downarrow$, respectively. This observation leads to Definition 1.1.

Example 2.4. Take the 2×2 matrix

$$A = \begin{pmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{pmatrix}.$$

Applying the collapsing operations, we get

$$\sigma_\downarrow(A) = \begin{pmatrix} a_{1,1} + a_{2,1} & a_{1,2} + a_{2,2} \end{pmatrix}, \qquad \sigma_\rightarrow(A) = \begin{pmatrix} a_{1,2} + a_{2,2} \\ a_{2,1} + a_{2,2} \end{pmatrix}, \qquad \text{and}$$

$$\sigma(A) = \begin{pmatrix} a_{1,1} + a_{1,2} + a_{2,1} + a_{2,2} \end{pmatrix}.$$

3. Equivalence of Gaussian blur and collapsing sum

In this section, we place the collapsing sum on a matrix-theoretic foundation and explicitly connect it with Gaussian blur via Theorem 3.12.

The following properties of the collapsing sum follow directly from Definition 1.1.

Proposition 3.1. *Let A and B be $m \times n$ matrices and c be any real number. Then*

(1) $\sigma(A + B) = \sigma(A) + \sigma(B)$,

(2) $\sigma(cA) = c \cdot \sigma(A)$,

(3) $\sigma_\downarrow(A^T) = \sigma_\rightarrow(A)^T$, and

(4) $\sigma(A^T) = \sigma(A)^T$

whenever the operations are defined. Moreover, the first two statements also hold for $\sigma_\downarrow$ and $\sigma_\rightarrow$.

Much of the investigation will examine repeated application of the collapsing sum. Let A be an $m \times n$ matrix. Then $\sigma^0(A) = A$, and for each positive integer $1 \le s < \min\{m, n\}$, we define $\sigma^s(A) = \sigma(\sigma^{s-1}(A))$. The operators $\sigma_\downarrow^s$ and $\sigma_\rightarrow^s$ are defined similarly.

Let I_m be the $m \times m$ identity matrix and $\delta_{i,j}$ be the Kronecker delta function

$$\delta_{i,j} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \ne j. \end{cases}$$

Definition 3.2. We denote by R_m the $(m-1) \times m$ matrix with entries $r_{i,j} = \delta_{i,j} + \delta_{i+1,j}$. For a positive integer $k < m$, we define $R_m^{\underline{k}}$ as the product $R_{m-k+1}R_{m-k+2}\cdots R_m$. Further, let $R_m^{\underline{0}} = I_m$.

The matrices R_m have 1's on the diagonal and superdiagonal and 0's elsewhere. The notation $R_m^{\underline{k}}$ is defined analogously to the falling power notation $n^{\underline{k}} = n(n-1)\cdots(n-k+1)$.

Example 3.3. We have

$$R_4 = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$

and

$$R_4^{\underline{2}} = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 1 & 0 \\ 0 & 1 & 2 & 1 \end{pmatrix}.$$

Proposition 3.4. *Let A be an $m \times n$ matrix with $m, n \geq 2$. Then $\sigma_{\downarrow}^s(A) = R_m^{\underline{s}} A$ and $\sigma_{\rightarrow}^s(A) = A(R_n^{\underline{s}})^T$.*

Proof. First note that $R_m A$ is an $(m-1) \times n$ matrix. Using the definition of R_m, the entry $(R_m A)_{i,j}$ is

$$\sum_{k=1}^{m} (\delta_{i,k} + \delta_{i+1,k}) a_{k,j} = a_{i,j} + a_{i+1,j} = \sigma_{\downarrow}(A)_{i,j}.$$

A quick induction argument shows that $\sigma_{\downarrow}^s(A) = R_m^{\underline{s}} A$. The calculation for the second assertion is similar. $\qquad\square$

Consequently, $\sigma^s(A) = (R_m^{\underline{s}}) A (R_n^{\underline{s}})^T$.

Proposition 3.5. *Let m be a positive integer and $s \leq m$ be a nonnegative integer. Then $R_m^{\underline{s}}$ is an $(m-s) \times m$ matrix with entries $(R_m^{\underline{s}})_{i,j} = \binom{s}{j-i}$.*

Proof. We proceed by induction. For $s = 0$, the theorem simplifies to the definition of $I_m = R_m^{\underline{0}}$. Now suppose that the theorem holds for some nonnegative integer k. Then $R_m^{\underline{k+1}} = R_{m-k} R_m^{\underline{k}}$. Since $R_m^{\underline{k}}$ is an $(m-k) \times m$ matrix, $R_m^{\underline{k+1}}$ is an $(m-k-1) \times m$ matrix. Further, by writing $(R_{m-k})_{i,r} = \delta_{i,r} + \delta_{i+1,r}$, we have

$$(R_m^{\underline{k+1}})_{i,j} = \sum_{r=1}^{m-k} (R_{m-k})_{i,r} (R_m^{\underline{k}})_{r,j}$$

$$= \binom{k}{j-i} + \binom{k}{j-(i+1)}$$

$$= \binom{k+1}{j-i},$$

so the formula holds by induction. $\qquad\square$

We now introduce an object that will facilitate the proof of Theorem 3.12.

Definition 3.6. Let $a < m$ and $b < n$ be nonnegative integers. The *coefficient matrix* $C_{m \times n}^{a,b} = (c_{i,j})$ is the unique $m \times n$ matrix such that $\sum_{i,j} \sigma_{\downarrow}^a \sigma_{\rightarrow}^b(A)_{i,j} = \sum_{i,j} c_{i,j} a_{i,j}$ for all $m \times n$ matrices A. We abbreviate $C_n^{a,b} := C_{n \times n}^{a,b}$ and $C_{m \times n}^a := C_{m \times n}^{a,a}$.

One interpretation of the coefficient matrix uses indeterminates. Let $X = (x_{i,j})$ be an $m \times n$ matrix of indeterminates; that is, the entries of X are distinct symbols, not numbers. The entry $c_{i,j}$ of the coefficient matrix $C_{m \times n}^{a,b}$ is the sum of the coefficients of $x_{i,j}$ across all entries of $\sigma_{\downarrow}^a \sigma_{\rightarrow}^b(X)$. Thus, one way to think of the coefficient matrix is that its (i,j)th entry represents the number of times that $x_{i,j}$ appears in $\sigma_{\downarrow}^a \sigma_{\rightarrow}^b(X)$.

We now work to describe the entries of the coefficient matrices explicitly.

Definition 3.7. Let A be an $m \times n$ matrix and $\mathbf{e}_n$ be the $n \times 1$ vector in which each entry is 1. The *column sum vector* of A is $\alpha = A^T \mathbf{e}_m$, and the *row sum vector* of A is $\beta = A\mathbf{e}_n$. That is, α_j is the sum of the elements in the jth column of A, and β_j is the sum of the elements of the jth row of A.

Lemma 3.8. *Let $X = (x_{i,j})$ be a matrix of indeterminates and A and B be matrices such that the product AXB is defined. If α is the column sum vector of A and β is the row sum vector of B, then the coefficient of $x_{p,q}$ in the formal expression $\sum_{i,j} (AXB)_{i,j}$ is $\alpha_p \beta_q$.*

Proof. Choose any indeterminate $x_{p,q}$. We have

$$\sum_{i,j} (AXB)_{i,j} = \sum_{i=1}^{m} \sum_{j=1}^{n} \left[\sum_{k=1}^{m} \sum_{r=1}^{n} a_{i,k} \cdot x_{k,r} \cdot b_{r,j} \right].$$

We obtain the coefficient of $x_{p,q}$ by summing only those terms where $k = p$ and $r = q$. This coefficient is thus

$$\sum_{i=1}^{m} \sum_{j=1}^{n} a_{i,p} b_{q,j} = \left[\sum_{i=1}^{m} a_{i,p} \right] \left[\sum_{j=1}^{n} b_{q,j} \right].$$

The left term in this product is α_p, and the right term is β_q. $\qquad\square$

Proposition 3.9. *Let α be the column sum vector of R_m^a and β be the column sum vector of R_n^b. Then $C_{m \times n}^{a,b} = \alpha \beta^T$.*

Proof. Let X be an $m \times n$ matrix of indeterminates. Apply Lemma 3.8 to $\sigma_{\downarrow}^a \sigma_{\rightarrow}^b (X) = (R_m^a) X (R_n^b)^T$. The row sum vector of $(R_n^b)^T$ is simply the column sum vector of R_n^b. The sum in Lemma 3.8 is the sum that defines the coefficient matrix, so the (i,j)th entry of the coefficient matrix $C_{m \times n}^{a,b}$ is $(\alpha \beta^T)_{i,j} = \alpha_i \beta_j$. $\qquad\square$

Proposition 3.9 implicitly gives the following formula for coefficient matrices.

Corollary 3.10. *Let m and n be positive integers and $a < m$ and $b < n$ be nonnegative integers. The coefficient matrix $C_{m \times n}^{a,b}$ has entries $c_{i,j} = \left[\sum_{\ell=1}^{m-a} \binom{a}{i-\ell} \right] \left[\sum_{\ell=1}^{n-b} \binom{b}{j-\ell} \right]$.*

Proof. Proposition 3.5 shows that

$$\alpha_i = \sum_{\ell=1}^{m-s} (R_m^a)_{\ell,i} = \sum_{\ell=1}^{m-a} \binom{a}{i-\ell}.$$

A similar calculation holds for β_j. $\qquad\square$

Corollary 3.11. *Let A be an $m \times n$ matrix. The value of the single entry of the matrix $\sigma_{\downarrow}^{m-1} \sigma_{\rightarrow}^{n-1}(A)$ is $\sum_{i=1}^{m} \sum_{j=1}^{n} \binom{m-1}{i-1} \binom{n-1}{j-1} a_{i,j}$.*

Proof. Let $C_{m \times n}^{m-1,n-1} = (c_{i,j})$. Since $\sigma_{\downarrow}^{m-1} \sigma_{\rightarrow}^{n-1}(A)$ has a single entry, we have

$$\sigma^{n-1}(A)_{1,1} = \sum_{i,j} \sigma^{n-1}(A)_{i,j} = \sum_{i,j} c_{i,j} a_{i,j}$$

by the definition of the coefficient matrix. Corollary 3.10 shows that $c_{i,j} = \binom{m-1}{i-1}\binom{n-1}{j-1}$. $\quad\square$

The entries of $\sigma^s(A)$ are determined by the blocks of A of size $(s+1)\times(s+1)$. From this observation, Corollary 3.11 can be used to find the value of any entry in $\sigma^s(A)$: simply apply the corollary to the submatrix $(a_{p+i,q+j})_{i,j=0}^s$ to determine the value of $\sigma^s(A)_{i,j}$.

Recall that to apply a kernel matrix, we need to specify an edge-handling technique, wherein we extend the matrix A to a matrix A'. Then applying the filter to A with the edge-handling technique is equivalent (by definition) to applying the filter to A' with cropping.

Theorem 3.12. *Suppose a matrix A and an edge-handling technique yielding the extension A' of A are given. Then $G_{2r+1} * A = 4^{-2r}\sigma^{2r}(A')$ for all nonnegative integers r.*

Proof. Each entry of $G_{2r+1} * A$ corresponds to a block of A' of size $(2r+1)\times(2r+1)$. From equation (1), the value of the entry $(G_{2r+1} * A)_{p,q}$ is

$$\frac{1}{4^{2r}} \sum_{i=-r}^{r} \sum_{j=-r}^{r} \binom{2r}{i+r}\binom{2r}{j+r} a'_{p+i,q+i}.$$

On the other hand, let $B := (a'_{p+i,q+j})_{i,j=-r}^{r}$ be a submatrix of A'. Applying Corollary 3.11 and then (1) gives

$$\sigma^{2r}(A')_{p,q} = \sigma^{2r}(B)_{1,1}$$
$$= \sum_{i=0}^{2r} \sum_{j=0}^{2r} \binom{2r}{i}\binom{2r}{j} b_{i,j}$$
$$= \sum_{i=-r}^{r} \sum_{j=-r}^{r} \binom{2r}{i+r}\binom{2r}{j+r} a'_{p+i,q+i}$$
$$= 4^{2r}(G_{2r+1} * A)_{p,q}. \qquad\square$$

Theorem 3.12 may be equivalently stated as an equality of operators:

$$G_{2r+1} = 4^{-2r}\sigma^{2r}.$$

4. Further properties of the collapsing sum

4.1. Special classes of matrices.

Definition 4.1. A *Toeplitz matrix* is an $m \times n$ matrix A with the property that $a_{i,j} = a_{k,\ell}$ whenever $i-j = k-\ell$. We denote by $\text{toep}(a_{-n+1},\dots,a_{m-1})$ the $m \times n$ Toeplitz matrix A with entries $a_{i,j} = a_{i-j}$.

Example 4.2. The general 4×5 Toeplitz matrix $\text{toep}(a_{-4}, \ldots, a_3)$ is

$$\begin{pmatrix} a_0 & a_{-1} & a_{-2} & a_{-3} & a_{-4} \\ a_1 & a_0 & a_{-1} & a_{-2} & a_{-3} \\ a_2 & a_1 & a_0 & a_{-1} & a_{-2} \\ a_3 & a_2 & a_1 & a_0 & a_{-1} \end{pmatrix}.$$

Toeplitz matrices have applications in a wide variety of pure and applied areas, including representation theory, signal processing, differential and integral equations, and quantum mechanics. Moreover, every $n \times n$ matrix can be decomposed as the product of at most $2n + 5$ Toeplitz matrices [4].

In what follows, we use $(m + 1) \times (n + 1)$ Toeplitz matrices $\text{toep}(a_{-n}, \ldots, a_m)$ to slightly simplify the statements of the results.

Proposition 4.3. *Let* $A = \text{toep}(a_{-n}, \ldots, a_m)$ *be an* $(m + 1) \times (n + 1)$ *Toeplitz matrix. Then* $\sum_{k=-n}^{m} \binom{m+n}{n+k} a_k$ *is the single entry of* $\sigma_\downarrow^m \sigma_\rightarrow^n(A)$.

Proof. We can decompose the Toeplitz matrix into "stripes":

$$A = \sum_{k=-n}^{m} \text{toep}(0, \ldots, 0, a_k, 0, \ldots, 0).$$

Since by Proposition 3.1 the collapsing sum distributes over addition, we need only consider the case when one a_k is nonzero. Moreover, since $\sigma(cA) = c\sigma(A)$, we can restrict to $a_k = 1$.

Therefore, suppose $A = \text{toep}(0, \ldots, 0, 1, 0, \ldots, 0)$, so that

$$a_{i,j} = \begin{cases} 1 & \text{if } i - j = k \\ 0 & \text{otherwise.} \end{cases}$$

We use the convention that $\binom{n}{r} = 0$ if $r < 0$ or $r > n$. Applying Corollary 3.11 and the binomial symmetry $\binom{n}{r} = \binom{n}{n-r}$ gives

$$\begin{aligned} \sigma_\downarrow^m \sigma_\rightarrow^n(A)_{1,1} &= \sum_{i=0}^{m} \sum_{j=0}^{n} \binom{m}{i}\binom{n}{j} a_{i+1,j+1} \\ &= \sum_{i=0}^{m} \binom{m}{i}\binom{n}{i-k} \\ &= \sum_{i=0}^{m} \binom{m}{i}\binom{n}{(n+k)-i}. \end{aligned}$$

Applying the well-known identity $\sum_{i=0}^{m} \binom{m}{i}\binom{n}{r-i} = \binom{m+n}{r}$ finishes the proof. $\square$

A direct application of Proposition 4.3 yields the following.

Corollary 4.4. *The single entry in* $\sigma^n(I_{n+1})$ *is the central binomial coefficient* $\binom{2n}{n}$.

A similar result holds for coefficient matrices.

Proposition 4.5. *The single entry in $\sigma^n(C_{n+1}^n)$ is $\binom{2n}{n}^2$.*

Proof. From Corollary 3.11, we have

$$\sigma^n(C_{n+1}^n) = \sum_{i=0}^{n} \sum_{j=0}^{n} \binom{n}{i}\binom{n}{j} c_{i,j} = \sum_{i=0}^{n} \sum_{i=0}^{n} \binom{n}{i}^2 \binom{n}{j}^2.$$

Using the identity $\sum_{k=0}^{n} \binom{n}{k}^2 = \binom{2n}{n}$ completes the proof. □

Recall that $J_{m \times n}$ denotes the $m \times n$ matrix with each entry equal to 1.

Proposition 4.6. *Let $a < m$ and $b < n$ be nonnegative integers. Then $\sum_{i,j}(C_{m \times n}^{a,b})_{i,j} = 2^{a+b}(m - a)(n - b)$.*

Proof. It follows from Definition 3.6 that the sum of the entries of $C_{m \times n}^{a,b}$ is equal to the sum of the entries of $\sigma_\downarrow^a \sigma_\rightarrow^b(J_{m \times n})$. Each $(a + 1) \times (b + 1)$ block of $J_{m \times n}$ is $J_{(a+1) \times (b+1)}$, so $\sigma_\downarrow^a \sigma_\rightarrow^b(J_{m \times n})_{i,j} = \sigma_\downarrow^a \sigma_\rightarrow^b(J_{(a+1) \times (b+1)})_{1,1}$ for all $1 \le i \le m - a$ and $1 \le j \le n - b$. Since $J_{(a+1) \times (b+1)}$ is a Toeplitz matrix, Proposition 4.3 gives

$$\sigma_\downarrow^a \sigma_\rightarrow^b(J_{(a+1) \times (b+1)})_{1,1} = \sum_{i=0}^{a+b} \binom{a+b}{i} = 2^{a+b}.$$

Noting that $\sigma_\downarrow^a \sigma_\rightarrow^b(J_{m \times n})$ has $(m - a)(n - b)$ entries completes the proof. □

The proof of Corollary 3.11 shows that the Gaussian blur kernel G_{2r+1} is proportional to the coefficient matrix C_{2r+1}^{2r}. Similarly, the box blur kernel B_{2r+1} is proportional to the coefficient matrix C_{2r+1}^0. The constant of proportionality in both cases is $(\sum_{i,j}(C_{2r+1}^s)_{i,j})^{-1}$, where $s = 2r$ or $s = 0$, respectively. Thus, the coefficient matrices are a generalization that unite these two filters. That is, the expression $(2^{a+b}(m - a)(n - b))^{-1} C_{m \times n}^{a,b}$ provides an interpolation between box blur and Gaussian blur.

4.2. Further connections with Gaussian blur.

Waltz and Miller [3] extend their techniques to non-square blurs, that is, Gaussian-like blurs using non-square kernel matrices. These can be defined in parallel to the square Gaussian blurs. If $G_{a \times b}$ denotes the kernel for the $a \times b$ Gaussian blur, then

$$(G_{a \times b})_{i,j} = 2^{-(a+b-2)} \binom{a-1}{i-1}\binom{b-1}{j-1}.$$

Since either a or b might be even, there may be no central element, so here we index from $(1,1)$ in the top left corner of the matrix.

The kernel matrix $G_{a \times b}$ is proportional to the coefficient matrix for a fully collapsed $a \times b$ matrix. This extends Theorem 3.12, since convolving $G_{a \times b}$ with a matrix A is equivalent to applying $2^{-(a+b-2)}\sigma_\downarrow^{a-1}\sigma_\rightarrow^{b-1}$ to the extended matrix A'.

The authors also venture into higher dimensions and discuss higher-dimensional blurs. We can easily transfer this idea to the language of the collapsing sum. Suppose we want to collapse (or, equivalently, blur) an n-dimensional array. We can define $\sigma_{\vec{i}}$, for $1 \le i \le n$,

to be the operator that "collapses" the array in the ith direction, akin to the effects of $\sigma_{\downarrow}$ and $\sigma_{\rightarrow}$ in two dimensions. Define $\sigma_n := \sigma_{\vec{1}} \cdots \sigma_{\vec{n}}$. Then powers of $2^{-n}\sigma_n$ give the higher-dimensional blur that Waltz and Miler describe. As before, general rectangular blurs are obtained by simply composing the operators $\frac{1}{2}\sigma_{\vec{i}}$ for various values of i.

4.3. **A generalized collapsing sum.** Waltz and Miller's algorithm for Gaussian blur may be extended to an operator that returns weighted sums of entries.

Definition 4.7. Let γ be an $b_1 \times b_2$ matrix and A be an $m \times n$ matrix with $m, n \geq \max\{b_1, b_2\}$. Then $\sigma_\gamma(A)$ is an $(m - b_1) \times (n - b_2)$ matrix with

$$\sigma_\gamma(A)_{p,q} = \sum_{i=0}^{b_1-1} \sum_{j=0}^{b_2-1} \gamma_{i+1,j+1} a_{p+i,q+j}.$$

If $\gamma = \left(\begin{smallmatrix} 1 & 1 \\ 1 & 1 \end{smallmatrix} \right)$, then we recover the original collapsing sum. Moreover, if $\gamma = (1\,1)$, then $\sigma_\gamma = \sigma_{\rightarrow}$, and $\sigma_{\gamma^T} = \sigma_{\downarrow}$.

For any matrix γ of rank 1, there exist two column vectors ρ and φ such that $\gamma = \rho\varphi^T$. Waltz and Miller's algorithm may be easily adapted for any 2×2 rank-1 matrix. Our previous results on the collapsing sum may also be extended to σ_γ for any (not necessarily square) matrix γ of rank 1.

Definition 4.8. Let φ be a column vector with k entries. Let φ be a column vector with k entries. The $(m - k + 1) \times m$ matrix R_m^φ has entries $(R_m^\varphi)_{p,q} = \sum_{i=0}^{k-1} \varphi_{i+1} \delta_{p+i,q}$.

Again, notice that if $\varphi = (1\,1)$, then $R_m^\varphi = R_m$. We define the falling powers of these matrices analogously to those of R_m. A generalized form of Proposition 3.9 holds in that, for any $m \times n$ matrix A,

$$\sigma_\rho^a \sigma_{\varphi^T}^b(A) = (R_m^\rho)^{\underline{a}} A [(R_n^{\varphi^T})^{\underline{b}}]^T.$$

In particular, if $\gamma = \rho\varphi^T$, then

$$\sigma_\gamma^s(A) = (R_m^\rho)^{\underline{s}} A [(R_n^{\varphi^T})^{\underline{s}}]^T.$$

Similar extensions may be obtained for other results, including the entries of the corresponding coefficient matrices.

5. Conclusion

By introducing the collapsing sum operators, we have provided a new combinatorial way to view Gaussian blur. We established the close connection between these concepts and also established a collection of theoretical results on the collapsing sum.

It would be interesting to study the collapsing sum as a matrix operator in its own right. For example, if G is an abelian group and $G^{m \times n}$ represents the additive group of $m \times n$ matrices with entries in G, then the collapsing sum is a map from $G^{m \times n}$ to $G^{(m-1) \times (n-1)}$. What are the combinatorial and algebraic properties of this map?

6. Acknowledgements

References

[1] Dimitrios Charalampidis. Recursive implementation of the Gaussian filter using truncated cosine functions. *Transactions on Signal Processing*, 64(14):3554–3565, 2016.

[2] Elhanan Elboher and Michael Werman. Efficient and accurate Gaussian image filtering using running sums. In *2012 12th International Conference on Intelligent Systems Design and Applications (ISDA)*, pages 897–902. IEEE, 2012.

[3] Frederick Waltz and John Miller. An efficient algorithm for Gaussian blur using finite-state machines. In *SPIE Conference on Machine Vision Systems for Inspection and Metrology VII*, pages 334–341, 1998.

[4] Ke Ye and Lek-Heng Lim. Every matrix is a product of Toeplitz matrices. *Foundations of Computational Mathematics*, 16:577–598, 2016.

Student biographies

Travis Dillon: (*Corresponding author:* travis.a.dillon@lawrence.edu) Travis is a third-year student at Lawrence University majoring in mathematics. He intends to pursue a doctorate in pure mathematics. When not engaged in math, he can often be found curled up with a book.