



WIKIPEDIA
The Free Encyclopedia

Main page
Contents
Featured content
Current events
Random article
Donate to Wikipedia
Wikipedia store

Interaction
Help
About Wikipedia
Community portal
Recent changes
Contact page

Tools
What links here
Related changes
Upload file
Special pages
Permanent link
Page information
Wikidata item
Cite this page

Print/export
Create a book
Download as PDF
Printable version

Languages
Čeština
Deutsch
Français
עברית
中文

Edit links

[Create account](#) [Log in](#)

Article [Talk](#)

[Read](#) [Edit](#) [View history](#)

Search

Summed area table

From Wikipedia, the free encyclopedia

A **summed area table** is a [data structure](#) and [algorithm](#) for quickly and efficiently generating the sum of values in a rectangular subset of a grid. In the [image processing](#) domain, it is also known as an **integral image**. It was first introduced to [computer graphics](#) in 1984 by [Frank Crow](#) for use with [mipmaps](#). In [computer vision](#) it was first prominently used within the [Viola–Jones object detection framework](#) in 2001.

However, historically, this principle is very well known in the study of multi-dimensional probability distribution functions, namely in computing 2D (or ND) probabilities (area under the probability distribution) from the respective [cumulative distribution functions](#).^[1]

Contents [\[hide\]](#)

- [1 The algorithm](#)
- [2 Extensions](#)
- [3 References](#)
- [4 External links](#)

The algorithm [\[edit\]](#)

As the name suggests, the value at any point (x, y) in the summed area table is just the sum of all the pixels above and to the left of (x, y) , inclusive:^{[2][3]}

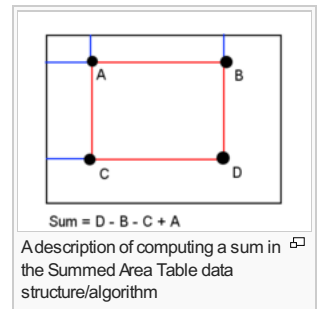
$$I_{\Sigma}(x, y) = \sum_{\substack{x' \leq x \\ y' \leq y}} i(x', y')$$

Moreover, the summed area table can be computed efficiently in a single pass over the image, using the fact that the value in the summed area table at (x, y) is just:

$$I(x, y) = i(x, y) + I(x - 1, y) + I(x, y - 1) - I(x - 1, y - 1)$$

Once the summed area table has been computed, the task of evaluating the intensities over any rectangular area requires only four array references. This allows for a constant calculation time that is independent of the size of the rectangular area. That is, using the notation in the figure at right, having $A=(x_0, y_0)$, $B=(x_1, y_0)$, $C=(x_0, y_1)$ and $D=(x_1, y_1)$, the sum of $i(x, y)$ over the rectangle spanned by A, B, C and D is:

$$\sum_{\substack{x_0 \leq x \leq x_1 \\ y_0 \leq y \leq y_1}} i(x, y) = I(D) + I(A) - I(B) - I(C)$$



Extensions [\[edit\]](#)

- This method is naturally extended to continuous domains.^[1]
- The method can be also extended to high-dimensional images.^[4] If the corners of the rectangle are x^p with p in $\{0, 1\}^d$, then the sum of image values contained in the rectangle are computed with the formula

$$\sum_{p \in \{0, 1\}^d} (-1)^{d - \|p\|_1} I(x^p)$$

where $I(x)$ is the integral image at x and d the image dimension. The notation x^p correspond in the example to $d = 2$: $A = x^{(0,0)}$, $B = x^{(1,0)}$, $C = x^{(0,1)}$ and $D = x^{(1,1)}$. In [neuroimaging](#), for example, the images have dimension $d = 3$ or $d = 4$, when using [voxels](#) or voxels with a time-stamp.

- This method has been extended to high-order integral image as in the work of Phan et al. ^[5] who provided two, three, or four integral images for quickly and efficiently calculating the standard deviation (variance), skewness, and kurtosis of local block in the image. This is detailed below.

To compute [variance](#) or [standard deviation](#) of a block, we need two integral images:

$$I(x, y) = \sum_{\substack{x' \leq x \\ y' \leq y}} i(x', y')$$
$$I^2(x, y) = \sum_{\substack{x' \leq x \\ y' \leq y}} i^2(x', y')$$

The variance is given by:

$$\text{Var}(X) = \frac{1}{n} \sum_{i=1}^n (x_i - \mu)^2.$$









Let S_1 and S_2 denote the summations of block $ABCD$ of I and I^2 , respectively. S_1 and S_2 are computed quickly by integral image. Now, we manipulate the variance equation as:

$$\text{Var}(X) = \frac{1}{n} \sum_{i=1}^n (x_i^2 - 2 \cdot \mu \cdot x_i + \mu^2) = \frac{1}{n} \left(\sum_{i=1}^n (x_i^2) - 2 \cdot \sum_{i=1}^n (\mu \cdot x_i) + \sum_{i=1}^n (\mu^2) \right) = \frac{1}{n} \left(\sum_{i=1}^n (x_i^2) - 2 \cdot \sum_{i=1}^n (\mu \cdot x_i) + n \cdot (\mu^2) \right)$$

Where $\mu = S_1/n$ and $S_2 = \sum_{i=1}^n (x_i^2)$

Similar to the estimation of the mean (μ) and variance (V_{arr}), which requires the integral images of the first and second power of the image respectively (i.e. I, I^2); manipulations similar to the ones mentioned above can be made to the third and fourth powers of the images (i.e. $I^3(x, y), I^4(x, y)$) for obtaining the skewness and kurtosis.^[5] But one important implementation detail that must be kept in mind for the above methods, as mentioned by F Shafait et al. ^[6] is that of integer overflow occurring for the higher order integral images in case 32-bit integers are used.

References [\[edit\]](#)

1. ^a ^b Finkelstein, Amir; neeratsharma (2010). "Double Integrals By Summing Values Of Cumulative Distribution Function" . *Wolfram Demonstration Project*.
2. ^a Crow, Franklin (1984). "Summed-area tables for texture mapping"  (PDF). *SIGGRAPH '84: Proceedings of the 11th annual conference on Computer graphics and interactive techniques*. pp. 207–212.
3. ^a Viola, Paul; Jones, Michael (2002). "Robust Real-time Object Detection"  (PDF). *International Journal of Computer Vision*.
4. ^a Tapia, Ernesto (January 2011). "A note on the computation of high-dimensional integral images". *Pattern Recognition Letters* **32** (2). doi:10.1016/j.patrec.2010.10.007 .
5. ^a ^b Phan, Thien; Sohoni, Sohumi; Larson, Eric C.; Chandler, Damon M. (22 April 2012). "Performance-analysis-based acceleration of image quality assessment"  (PDF). *2012 IEEE Southwest Symposium on Image Analysis and Interpretation*: 83. doi:10.1109/SSIAI.2012.6202458 .
6. ^a Shafait, Faisal; Keyzers, Daniel; M. Breuel, Thomas (January 2008). "Efficient implementation of local adaptive thresholding techniques using integral images"  (PDF). *Electronic Imaging*. doi:10.1117/12.767755 .

External links [[edit](#)]

- Summed table implementation in object detection [🔗](#)

Lecture videos

- An introduction to the theory behind the integral image algorithm [↗](#)
- A demonstration to a continuous version of the integral image algorithm, from the Wolfram Demonstrations Project [↗](#)

Categories: Digital geometry | Computer graphics data structures

This page was last modified on 15 August 2015, at 08:48.

Text is available under the [Creative Commons Attribution-ShareAlike License](#); additional terms may apply. By using this site, you agree to the [Terms of Use](#) and [Privacy Policy](#). Wikipedia® is a registered trademark of the [Wikimedia Foundation, Inc.](#), a non-profit organization.

[Privacy policy](#) [About Wikipedia](#) [Disclaimers](#) [Contact Wikipedia](#) [Developers](#) [Mobile view](#)

