Joint integral histograms and its application in stereo matching — **Source link** ⧉

Ke Zhang, Gauthier Lafruit, Rudy Lauwereins, Luc Van Gool

Institutions: Katholieke Universiteit Leuven

Related papers:

- A taxonomy and evaluation of dense two-frame stereo correspondence algorithms

- Integral histogram: a fast way to extract histograms in Cartesian spaces

- Fast cost-volume filtering for visual correspondence and beyond

- Real-time spatiotemporal stereo matching using the dual-cross-bilateral grid

- Constant Time Stereo Matching

JOINT INTEGRAL HISTOGRAMS AND ITS APPLICATION IN STEREO MATCHING

Ke Zhang[1,2,4,*], *Gauthier Lafruit*[1], *Rudy Lauwereins*[1,2], *Luc Van Gool*[2,3]

imec, Kapeldreef 75, B-3001 Leuven, Belgium[1]
Katholieke Universiteit Leuven, Leuven, Belgium[2]
ETH Zurich, Zurich, Switzerland[3]
IBBT, Belgium[4]

ABSTRACT

In this paper, we first propose a technique, referred as *joint integral histograms*, for weighted filtering with O(1) computational complexity. The technique is built on the classic integral images and the recent integral histograms. In a joint integral histogram, instead of remembering bin occurrences, the value at each bin indicates an integral defined by two signals. Beyond the integral histograms, our method supports weighted filtering with a more general form, where the weight could be a function of a signal different from the signal to be filtered. Then, we present a local stereo matching approach as an instantiation of the technique. Using the joint integral histograms, we achieve a speedup factor of about two orders of magnitude. Thanks to the huge speedup, the stereo method is among the best local approaches in terms of the trade-off between matching accuracy and execution speed. Experimental results demonstrate the advantages of both the joint integral histograms technique and the stereo matching approach.

Index Terms— joint integral histograms, weighted filtering, O(1) computational complexity, stereo matching

1. INTRODUCTION

Filtering is a fundamental image processing task, which replaces a pixel by a uniform or weighted average of its neighbors. Its computational complexity normally depends on the size and the complexity of the filter. However, for uniform box filtering, the *integral images* [1] [2] reduce the computational complexity to O(1) regardless of the size of the filter. With one pre-computed integral image, evaluation of an arbitrary rectangular region can be completed with several arithmetic operations. The elegant technique was successfully adopted by many algorithms [3] [4].

Hussein et al. [5] extended integral images to a class of non-uniform filters, by expressing the weight function as multiple region-independent weight functions. It is suitable to accelerate linear filtering such as bilinear interpolation and spatial Gaussian weighting. Porikli [6] proposed the *integral histograms*, which generalizes the integral images to extract histogram features. Similarly to the integral images, integral histograms exploits the spatial arrangement of data points to build an integral histogram. With the integral histogram, histogram of any rectangular regions can be computed in constant time. The technique was successfully applied to accelerate the non-linear bilateral filter [7] in [8], achieving a speedup factor of several orders of magnitude for large filters.

Inspired by integral images and integral histograms, we propose a more generalized technique and refer it as *joint integral his-

---

*Email: zhangke@imec.be

tograms* (JIH). Beyond the integral histograms, our method supports weighted filtering with a more general form, where the weight could be a function of a signal different from the signal to be filtered. In a joint integral histogram, the value at each bin indicates an integral defined by two signals, instead of remembering bin occurrences in a integral histogram. From this point of view, JIH can be regarded as a combination of integral images and integral histograms. Once a joint integral histogram is built up, filtering over any rectangular region can be performed with several arithmetic operations regardless of the region size. With JIH, huge speedup could be achieved for a broad class of weighted filtering.

To demonstrate the effectiveness of the JIH, we also present a stereo matching algorithm as an instantiation of it. Stereo matching is an important computer vision task with [9] as a substantial survey. Local stereo methods improve matching accuracy by appropriately aggregating support over local regions. Tombari et al. [10] evaluated many cost aggregation methods in a Winner-Takes-All framework, considering both matching accuracy and execution speed. Adaptive weight method [11] is one leading method in the evaluation in terms of accuracy. When aggregating cost over support regions, the method assigns an adaptive weight to each pixel. It greatly improves the matching accuracy, yet is very computationally expensive.

We propose a stereo matching algorithm, inspired by the algorithm [11]. The cost aggregation is formulated as a weighted filtering of the raw matching cost, where the weight depends on input stereo images. By applying the proposed JIH, we can achieve a speedup factor of two orders of magnitude. Experimental results show that our stereo matching approach is among the best local approaches in terms of the trade-off between matching accuracy and execution speed. The rest of the paper is organized as follows. Section 2 introduces the proposed JIH technique. The stereo matching method is presented in Section 3. Section 4 demonstrates the experimental results and Section 5 concludes the paper.

2. JOINT INTEGRAL HISTOGRAMS FOR O(1) WEIGHTED FILTERING

Our JIH is inspired by the integral images and integral histograms. Integral images is commonly used to execute box filtering with constant time. To sum the samples $f(x, y)$ in arbitrary rectangular regions $R = \{(x, y) | x \in (x_1, x_2], y \in (y_1, y_2]\}$, integral images takes two steps. First, an integral image $f^{\int}(x, y)$ is built as Eq.(1).

$$
\begin{aligned}
f^{\int}(x, y) = \sum_{x' \leq x, y' \leq y} f(x', y') = f(x, y) + f^{\int}(x - 1, y) \\
+ f^{\int}(x, y - 1) - f^{\int}(x - 1, y - 1)
\end{aligned}
\tag{1}
$$

$f^{\int}(x,y)$ is the sum of the samples positioned to the left and above of point (x,y) (including (x,y)). It can be iteratively built as Eq.(1) with $f(-1,0) = f(0,-1) = f(-1,-1) = 0$. Once the integral image is built, accumulation over R can be computed with 3 arithmetic operations as Eq.(2), regardless of the region size $\|R\|$.

$$\sum_{(x,y)\in R} f(x,y) = f^{\int}(x_2,y_2) - f^{\int}(x_1,y_2) - f^{\int}(x_2,y_1) \\ + f^{\int}(x_1,y_1) \quad (2)$$

Utilizing the concept of integral images, integral histograms was proposed to extract histograms over rectangular regions, also in constant time. First, an integral histogram $H^{\int}(x,y,b)$ is built as Eq.(3) at each point (x,y), where $\delta(.)$ is an impulse function.

$$H^{\int}(x,y,b_i) = \sum_{x'\leq x, y'\leq y} \delta(f(x',y') - b_i) \quad (3)$$

Its value at the ith bin b_i $H^{\int}(x,y,b_i)$ is the number of occurrences of the bin in the region to the left and above of the point (x,y) (including (x,y)). With the integral histograms, the value at one bin of the histograms over any rectangular region R can be obtained with 3 arithmetic operations Eq.(4), similarly to the integral images.

$$H_{b_i}^R = \sum_{(x,y)\in R} H(x,y,b_i) = H^{\int}(x_2,y_2,b_i) \\ - H^{\int}(x_2,y_1,b_i) - H^{\int}(x_1,y_2,b_i) + H^{\int}(x_1,y_1,b_i) \quad (4)$$

Integral histograms was successfully used to accelerate the bilateral filtering [8]. For the filters with the form of Eq.(5), with R_p a rectangle centered at point p, integral histograms is applicable when $g(.) = f(.)$.

$$f^\omega(p) = \frac{1}{W} \sum_{q\in R_p} \omega(g(p),g(q))f(q), W = \sum_{q\in R_p} \omega(g(p),g(q)) \quad (5)$$

Otherwise, it cannot be directly applied. To accelerate the more complex filtering with $g(.)$ different from $f(.)$, we propose the JIH technique as follows.

We transform $f^w(p)$ to the form in Eq.(6), where G is the set including all possible values of $g(.)$.

$$f^\omega(p) = \frac{\sum_{\hat{g}\in G}\left(\omega(g(p),\hat{g})\sum_{s\in R_p} f(s)\delta(g(s)-\hat{g})\right)}{\sum_{\hat{g}\in G}\left(\omega(g(p),\hat{g})\sum_{s\in R_p}\delta(g(s)-\hat{g})\right)} \quad (6)$$

Integral histograms is applicable to compute the denominator. First, an integral histogram is built for the image $g(.)$ as Eq.(3). Then, $\sum_{s\in R_p}\delta(g(s)-\hat{g}) = H_{\hat{g}}^{R_p}$ can be extracted in constant time using Eq.(4). However, since the numerator contains two different functions, we cannot directly get the numerator utilizing the integral histogram.

To facilitate the computation of the numerator, we construct a *joint integral histogram* $\mathbb{H}^{\int}(.)$ as Eq.(7).

$$\mathbb{H}^{\int}(x,y,b_i) = \sum_{x'\leq x, y'\leq y}\delta(g(x',y')-b_i)f(x',y') \quad (7)$$

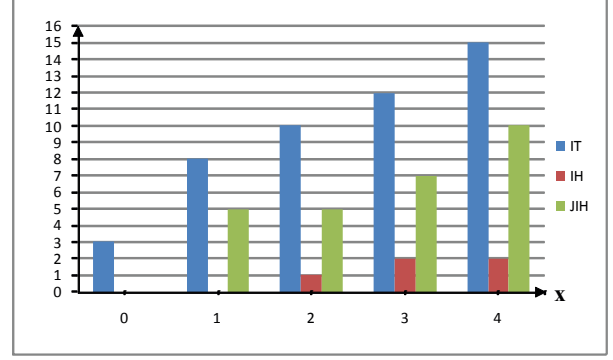The contribution of (x,y) to the joint integral histogram is jointly

Fig. 1. The value of an integral image (IT) $f^{\int}(x)$, an integral histogram (IH) $H^{\int}(x,b_i = 2)$ and a joint integral histogram (JIH) $\mathbb{H}^{\int}(x,b_i = 2)$ in a one-dimensional example, with $\{f(x)|x \in [0,4]\} = \{3,5,2,2,3\}$ and $\{g(x)|x \in [0,4]\} = \{1,2,4,2,2\}$. While IT and IH only depend on $\{f(.)\}$, JIH depends on both $\{f(.)\}$ and $\{g(.)\}$.

determined by both function $f(.)$ and function $g(.)$, i.e., $g(x,y)$ determines which bin to increase and $f(x,y)$ determines the value to increase at that bin. Comparing Eq.(7) to Eq.(1) and Eq.(3), the joint integral histogram can be regarded as a combination of an integral image and an integral histogram. Fig.1 is an one-dimensional example to illustrate their relation and difference. In the example, $\{f(x)|x \in [0,4]\}$ is an ordered set $\{3,5,2,2,3\}$ and $\{g(x)|x \in [0,4]\}$ is an ordered set $\{1,2,4,2,2\}$. The figure shows the value of an integral image $f^{\int}(x)$, the value of an integral histogram $H^{\int}(x,b_i = 2)$ and a joint integral histogram $\mathbb{H}^{\int}(x,b_i = 2)$.

Eq.(6) can be transformed to Eq.(8).

$$f^\omega(p) = \frac{\sum_{\hat{g}\in G}\left(\omega(g(p),\hat{g})\mathbb{H}_{\hat{g}}^{R_p}\right)}{\sum_{\hat{g}\in G}\left(\omega(g(p),\hat{g})H_{\hat{g}}^{R_p}\right)} \approx \frac{\sum_{i\in[0,\|B\|)}\left(\omega(g(p),b_i)\mathbb{H}_{b_i}^{R_p}\right)}{\sum_{i\in[0,\|B\|)}\left(\omega(g(p),b_i)H_{b_i}^{R_p}\right)} \quad (8)$$

With the joint integral histogram, $\mathbb{H}_{\hat{g}}^{R_p}$ is computed with 3 arithmetic operations as Eq.(4), regardless of the region size $\|R_p\|$. Therefore, the execution time of the weighted filtering is reduced to constant time, while the complexity of the direct method quadratically increases with the region radius. As indicated in [8], to obtain further speedup without sacrificing much accuracy, G could be quantized or mapped to a sparser set $B = \{b_i\}$ with $\|B\| < \|G\|$. Then, $f^\omega(p)$ is expressed in a more explicit form in Eq.(8).

3. FAST STEREO MATCHING USING JOINT INTEGRAL HISTOGRAMS

In this section, we propose a fast local stereo matching algorithm as an instantiation of the JIH technique. As an effective technique of deriving 3D information, stereo matching computes a disparity map D from a pair of stereo images, i.e. the left image I and the right image I'. For each pixel $p = (x_p, y_p)$ in I, a set of disparity hypothesis $\{d|d \in [0, d_{max}]\}$ is evaluated with the matching cost between p and its correspondence $p' = (x_p - d, y_p)$ in I'. In a Winner-Takes-All (WTA) framework, the disparity which produces
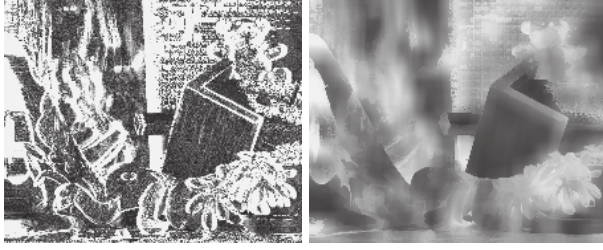
Fig. 2. (a) The raw matching cost $C_d(p)$ for the *Teddy* image at $d = 30$. (b) Filtered cost image $C_d^{\omega}(p)$ with the JIH using 8 bins. (The images are scaled for better visualization.)

the smallest matching cost is selected as the optimal disparity $D(p)$ for p.

The raw (per-pixel) matching cost $C_d(p)$ is normally defined based on the intensity difference between p and p'. Since the raw matching cost is lack of discriminative power, local stereo algorithms improve matching accuracy by aggregating the raw matching cost over local support regions, e.g. box filter method [9] as the simplest approach. Ideally, matching cost should be aggregated only from pixels with similar disparity values. A common assumption is that neighboring pixels with similar intensity are likely from the same object structure and therefore have similar disparity values. Based on the assumption, adaptive weight method [11] adaptively aggregates raw matching cost. For each pixel q in the support window of p, Yoon et al. [11] adapt the support weight of $C_d(q)$ according to the intensity distance and spatial distance between q and p. Though the method is capable of producing high accurate disparity maps, it is quite computationally expensive.

Our stereo matching algorithm is inspired by the adaptive weight method [11] and illustrated as follows. At the disparity d, the raw matching cost $C_d(p)$ is calculated with a truncated SAD method Eq.(9), where T is a truncation value.

$$C_d(p) = \min\{|I(p) - I'(p')|, T\} \qquad (9)$$

For color images, the cost is the sum of the differences in 3 channels. Then, we filter the cost image $C_d(.)$ with a weighted filter. Based on the assumption above, the contribution from q in R_p to p is weighted by their intensity difference in the input image I as Eq.(10).

$$C_d^{\omega}(p) = \sum_{q \in R_p} \omega(I(p), I(q))C_d(q), \omega(I(p), I(q)) = e^{-\frac{|I(p)-I(q)|}{\gamma}}$$
$$(10)$$

It assigns a higher weight to the pixels with smaller intensity difference to the anchor pixel p. Since the total weight for p across different disparities is the same, normalization is not necessary. Finally, the WTA method is used to select an optimal disparity $D(p)$ for each pixel p based on the filtered matching cost $C_d^{\omega}(p)$ as Eq.(11).

$$D(p) = \arg\min_d C_d^{\omega}(p) \qquad (11)$$

To compute the weight $\omega(I(p), I(q))$, a quantized version of I is used. According to our experimental results, 8 bins are enough to achieve similar accuracy as using the original image (256 bins). Fig. 2 shows the raw matching cost and the filtered cost image for the *Teddy* image [12] at $d = 30$. The filter radius is set to 15 and an image with 8 bins is used to calculate the weight $\omega(.)$. We can see that the noisy image of raw matching cost is smoothed while the object

boundaries are well preserved. It indicates that the raw matching cost is propagated mostly among pixels with similar disparity values and therefore the reliability of the matching cost is desirably increased.

Using the direct computing method, the computational complexity for filtering one image is $\mathrm{O}(M \times \|R_p\|)$, where M is the resolution of the input image. Though implemented in a speed-oriented method with a memory consumption of $\|R_p\|$ images (one weight window at each pixel), it still needs around 1s for filtering one cost image ($M = 450 \times 375$) at each disparity. Observing that Eq.(10) is a weighted filter similar to Eq.(5), we can use the proposed JIH to accelerate the filtering by replacing $f(.)-> C_d(.)$ and $g(.)-> I(.)$. We first build a joint integral histogram $\mathbb{H}^{\int}$ as Eq.(12).

$$\mathbb{H}^{\int}(x, y, b_i) = \sum_{x' \leq x, y' \leq y} \delta(I(x', y') - b_i)C_d(x', y') \qquad (12)$$

With $\mathbb{H}_{b_i}^{R_p}$ computed using 3 arithmetic operations as Eq.(4), $C_d^{\omega}(p)$ can be obtained in constant time as Eq.(13).

$$C_d^{\omega}(p) = \sum_{i \in [0, \|B\|)} \left(\omega(I(p), b_i)\mathbb{H}_{b_i}^{R_p} \right) \qquad (13)$$

By using the JIH, the time of filtering one image is reduced to 25ms, achieving a speedup factor of 40x. In addition, the memory consumption is reduced to $\|B\| = 8$ images for storing the joint integral histogram, compared with $\|R_p\| = 961$ images in the direct method.

4. EXPERIMENTAL RESULTS

We first evaluate the speedup achieved using the proposed JIH technique. The stereo matching algorithm is implemented using the direct method and the JIH method respectively. The computing platform is Intel Core Duo 2.66GHz. We test the execution time on the *Teddy* image, which has a image resolution of 450×375 and a disparity range of 60. The execution time is shown in Fig.3, depending on the radius of the filter. The time of direct method quadratically increases with the radius. In comparison, the execution time with the JIH is constant, i.e., 1.5s for 8 bins and 2.8s for 16 bins, regardless of the radius. For modest region sizes, the JIH method could be faster than the direct method in about two orders of magnitude.

Then we evaluate the accuracy of the proposed stereo matching approach, with the Middlebury stereo benchmark [12]. The parameters remain the same for all the tested images, i.e., 8 bins for calculating the weight, truncation value of raw matching cost $T = 60$, $\gamma = 32$, and filter radius $r = 10$. Fig.4 shows the produced disparity maps for *Teddy* and *Cones* images. Thanks to the adaptive weight method, the object boundaries are well preserved. Table 1 illustrates the quantitative evaluation using [12]. It reports the error rates in non-occluded regions (Nocc) and depth-discontinuity regions (Disc), where the disparity error is bigger than 1.

We also compare the proposed method to other local methods reported in the Table 1 of [10]. In [10], about 20 cost aggregation methods are evaluated in a typical WTA framework without pre-processing or post-processing. In terms of speed, our method ranks 2nd, only behind the simplest fixed window method. Though our method is less accurate than [11], our method is faster in three orders of magnitude, i.e., 1.5s compared with 18 minutes. Compared to many other low-complexity methods [13] [14] [15], our method is both more accurate and faster. With an average rank 3.55 as the highest among all the methods, our method obtains an optimal point in terms of the trade-off between accuracy and speed.

Table 1. Evaluation results for the Middlebury stereo database [12] in terms of both matching accuracy and execution time (on *Teddy*) [10].

| Algorithm	*Tsukuba*		*Venus*		*Teddy*		*Cones*		Accuracy rank	Time mm:ss	Time rank	Average rank
	Nocc	Disc	Nocc	Disc	Nocc	Disc	Nocc	Disc				
Adapt. Weight	4.66	8.25	4.61	13.30	12.70	22.40	5.50	11.90	2.50	18:14	15	8.75
Ours	**6.21**	**25.70**	**4.98**	**27.00**	**14.00**	**29.90**	**6.92**	**17.80**	**5.11**	**$\sim$ 00:01**	**2**	**3.55**
Seg. based [13]	8.18	18.77	8.06	20.85	15.78	29.66	13.22	24.55	8.13	00:02	3	5.57
Mult. Win.(9W) [14]	8.51	27.59	6.47	34.30	17.57	38.04	10.75	26.60	11.25	00:13	7	9.12
Grad. Guided [15]	7.51	16.20	13.07	33.10	20.39	32.82	13.67	25.60	11.38	00:03	4	7.69
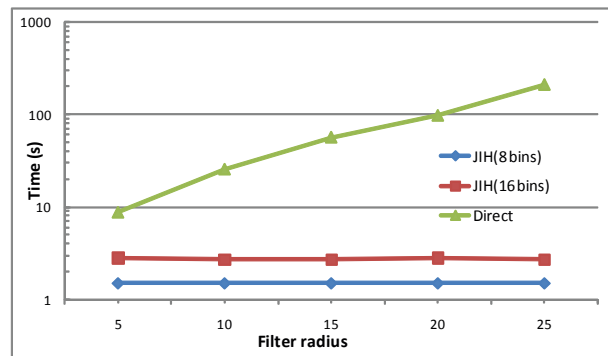
Fig. 3. Execution time of the proposed stereo matching algorithm for the *Teddy* image (in log scale), implemented using the direct method (direct) and the proposed joint integral histograms (JIH) respectively.

Fig. 4. (Top row) *Teddy* and *Cones* images and (bottom row) the produced disparity maps with the proposed stereo method.

5. CONCLUSIONS

In this paper, we proposed a joint integral histograms (JIH) technique. It can be viewed as a combination or generalization of the integral images technique and the integral histograms. Beyond the integral histograms technique, it supports a class of weighted filtering with constant time, where the weight can depend on a signal different from the signal to be filtered. The technique was instantiated into a local stereo matching algorithm. With the JIH, we achieved a speedup factor of about two orders of magnitude. The stereo method is among the best local methods in terms of the trade-off between matching accuracy and execution speed. Experimental results have validated the effectiveness of the JIH, which could be embedded into other applications.

6. REFERENCES

[1] F. Crow, "Summed-area tables for texture mapping," in *Proc. ACM SIGGRAPH*, 1984, pp. 207–212.

[2] P. Viola and M. Jones, "Robust real-time face detection," in *ICCV*, 2001, p. II: 747.

[3] H. Bay, T. Tuytelaars, and L. V. Gool, "Surf: Speeded up robust features," in *ECCV*, 2006.

[4] O. Tuzel, F. Porikli, and P. Meer, "Region covariance: A fast descriptor for detection and classification," in *ECCV*, 2006.

[5] M. Hussein, F. Porikli, and L. Davis, "Kernel integral images: A framework for fast non-uniform filtering," in *CVPR*, 2008.

[6] F. Porikli, "Integral histogram: A fast way to extract histograms in cartesian spaces," in *CVPR*, 2005.

[7] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *ICCV*, 1998.

[8] F. Porikli, "Constant time o(1) bilateral filtering," in *CVPR*, 2008.

[9] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *IJCV*, vol. 47, no. 1, pp. 7–42, May 2002.

[10] F. Tombari, S. Mattoccia, and L. Stefano, "Classification and evaluation of cost aggregation methods for stereo correspondence," in *IEEE CVPR*, 2008.

[11] K. J. Yoon and I. S. Kweon, "Adaptive support-weight approach for correspondence search," *IEEE PAMI*, vol. 28, no. 4, pp. 650–656, April 2006.

[12] "Middlebury stereo vision page," http://vision.middlebury.edu/stereo/.

[13] M. Gerrits and P. Bekaert, "Local stereo matching with segmentation-based outlier rejection," in *Conf. Computer and Robot Vision*, 2006.

[14] H. Hirschmuller, P. Innocent, and J. Garibaldi, "Real-time correlation-based stereo vision with reduced border errors," in *IJCV*, 2002.

[15] M. Gong and R. Yang, "Image-gradient-guided real-time stereo on graphics hardware," in *3DIM*, 2005.