

# CWMA: Circular Window Matching Algorithm

Daniel Miramontes-Jaramillo<sup>1</sup>, Vitaly Kober<sup>1</sup>, and Víctor Hugo Díaz-Ramírez<sup>2</sup>

<sup>1</sup> CICESE, Ensenada, B.C. 22860, México  
dmiramon@cicese.edu.mx, vkober@cicese.mx  
<sup>2</sup> CITEDI-IPN, Tijuana, B.C. 22510, México  
vhdiacz@citedi.mx

**Abstract.** Various vision applications exploit matching algorithms to locate a target object in a scene image. A new fast matching algorithm based on recursive calculation of oriented gradient histograms over several circular sliding windows is presented. In order to speed up the algorithm pyramidal image decomposition technique and parallel implementation with modern multicore processors are utilized. The proposed fast algorithm yields a good invariance performance for both in-plane and out-of-plane rotations of a scene image. Computer results obtained with the proposed algorithm are presented and compared with those of common algorithms in terms of matching accuracy and processing time.

## 1 Introduction

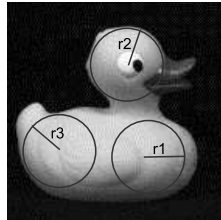
Recently numerous matching algorithms using features or keypoints were proposed. Among them, Scale Invariant Feature Transform (SIFT) [1] and Speeded-Up Robust Features (SURF) [2] are the most popular algorithms. These basic algorithms and their variants [3,4,5] can be used as references for comparison with new matching methods. Although feature-based matching methods are popular, template matching algorithms are an attractive alternative for real-time applications [6,7]. Template matching filters possess a good formal basis and can be implemented by exploiting massive parallelism in hybrid optodigital systems [6,8] or in high-performance digital hardware such as graphic processing units (GPU) [7,9] or field programmable gate arrays (FPGA) [10] at high speed. Another approach is a combination of feature-based and template matching algorithms. For example, Scale Invariant Compressed Histogram Transform (SICHT) [11] uses the Histograms of Oriented Gradients (HoG) [12] calculated in a moving window as features. In this paper we present a fast hybrid algorithm for a reliable matching that recursively calculates the histograms of oriented gradients in several sliding circular windows. The shape of sliding windows helps us to obtain a pretty good invariance to in-plane/out-of-plane image rotations with a slight scaling. The algorithm can be easily implemented using modern technology of multi-core processors. The performance of the proposed algorithm in a test database is compared with that of SIFT and SURF algorithms in terms of matching accuracy and processing time.

## 2 Proposed Approach

First, let us define a set of circular windows  $\{W_i, i = 1, \dots, M\}$  in a reference image as a set of the following closed disks:

$$W_i = \left\{ (x, y) \in \mathbb{R}^2 : (x - x_i)^2 + (y - y_i)^2 \leq r_i^2 \right\}, \quad (1)$$

where  $(x_i, y_i)$  are coordinates of the center and  $r_i$  is the radius of the  $i$ th disk. The circular windows with possible overlapping fill inside an object of interest in the reference image as shown in Fig. 1. The disks form a geometric structure that runs across a scene image. The histograms of oriented gradients are calculated in circular areas and further used for matching. It is interesting to note that at any position of the structure each disk contains image area that is unchangeable during rotation; therefore, the histogram of oriented gradients computed in a circular window is also invariant to rotation. Moreover, in order to obtain a high accuracy of matching relative positions of the disks described by the center distances and center-to-center angles should be taken into account. It is recommendable to choose a minimum number of equal disks with a radius to fill inside as much as possible the reference object. Actually, numerous experiments have showed that the number  $M$  of circular windows may be chosen from 2 to 4 to yield the matching performance comparable with that of the SIFT.



**Fig. 1.** Set of circular windows calculated from an object

Histograms of Oriented Gradients are good features for matching [12] because they possess a good discriminant capability and robust to small image deformations such as rotation and scale. First, at each position of the  $i$ th circular window on a scene image we compute gradients inside the window with the help of the Sobel operator [13]. Next, using the gradient magnitudes  $\{Mag_i(x, y) : (x, y) \in W_i\}$  and orientation values quantized for  $Q$  levels  $\{\varphi_i(x, y) : (x, y) \in W_i\}$ , the histogram of oriented gradients can be computed as follows:

$$HoG_i(\alpha) = \begin{cases} \sum_{(x,y) \in W_i} \delta(\alpha - \varphi_i(x, y)), & Mag_i(x, y) \geq Med \\ 0, & otherwise, \end{cases} \quad (2)$$

where  $\alpha = \{0, \dots, Q - 1\}$  are histogram values (bins),  $Med$  is the median value of image pixels inside of the circular window, and  $\delta(z) = \begin{cases} 1, & z = 0 \\ 0, & otherwise \end{cases}$  is the Kronecker delta function. Note that the calculation in Eq.(2) requires approximately  $\lceil \pi r_i^2 \rceil$  addition operations. In order to reduce computational complexity the calculation of the histograms at the sliding window position  $k$  can be performed in a recursive manner as follows:

$$\begin{aligned}
 HoG_i^k(\alpha) = HoG_i^{k-1}(\alpha) - \sum_{(x,y) \in OutP_i^{k-1}} \delta(\alpha - Out\varphi_i^{k-1}(x,y)) & \quad (3) \\
 + \sum_{(x,y) \in InP_i^k} \delta(\alpha - In\varphi_i^k(x,y)) &
 \end{aligned}$$

where  $OutP_i^{k-1}$  is a set of outgoing orientation values whose pixels belong to the half of the perimeter of the sliding window at step  $k - 1$ , that is,

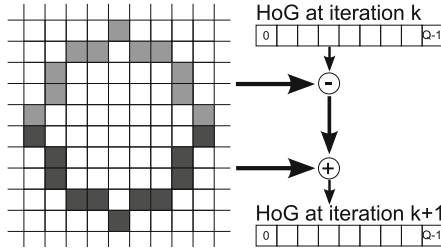
$$\left\{ \begin{aligned}
 Out\varphi_i^{k-1}(x,y) = \begin{cases} \varphi_i^{k-1}(x,y), & Mag_i^{k-1}(x,y) \geq Med^{k-1} \\ 0, & otherwise \end{cases} : & \quad (4) \\
 (x,y) \in OutP_i^{k-1} \end{aligned} \right\};$$

$InP_i^k$  is a set of incoming orientation values whose pixels belong to the half of the perimeter of the sliding window at step  $k$  given by

$$\left\{ \begin{aligned}
 In\varphi_i^k(x,y) = \begin{cases} \varphi_i^k(x,y), & Mag_i^k(x,y) \geq Med^k \\ 0, & otherwise \end{cases} : & \quad (5) \\
 (x,y) \in InP_i^k \end{aligned} \right\}.$$

The computational complexity of this calculation is approximately  $\lceil 2\pi r_i \rceil$  addition operations. Fig. 2 shows the recursive update of the histogram along columns. The recursive calculation can be used along columns as well as along rows. To provide rotation invariance a cyclic shift of the histogram moving a dominant orientation was proposed [11]. However, if there are several dominant orientations in the histogram owing to noise, this method does not work properly.

Another drawback is the method does not take into account scale invariance. To overcome these disadvantages, we utilize a normalized correlation operation for comparison of the histograms of the reference and scene images. Let us compute a centered and normalized histogram of oriented gradients of the reference as follows:



**Fig. 2.** Recursive histogram update

$$\overline{HoG_i^R}(\alpha) = \frac{HoG_i^R(\alpha) - Mean^R}{\sqrt{Var^R}}, \tag{6}$$

where  $Mean^R$  and  $Var^R$  are sample mean and variance of the histogram, respectively.

The correlation output for the  $i$ th circular window at position  $k$  can be computed with the help of the fast Inverse Fourier Transform [13] as follows:

$$C_i^k(\alpha) = IFT \left[ \frac{HS_i^k(\omega) HR_i^*(\omega)}{\sqrt{\sum_{q=0}^{Q-1} (HoG_i^k(q))^2 - (HS_i^k(0))^2}} \right], \tag{7}$$

where  $HS_i^k(\omega)$  is the Fourier Transform of the histogram of oriented gradients inside of the  $i$ th circular window over the scene image and  $HR_i(\omega)$  is the Fourier Transform of  $\overline{HoG_i^R}(\alpha)$ ; the asterisk denotes complex conjugate. The correlation peak is a measure of similarity of the two histograms, which can be obtained as follows:

$$P_i^k = \max_{\alpha} \{C_i^k(\alpha)\}. \tag{8}$$

The correlation peaks are in the range of  $[-1, 1]$ . It is of interest to note that the normalized correlation peaks possess two important properties: first, invariance to rotation because a cyclic shift of the histogram values corresponds to a cyclic shift of the correlation output and does not change the correlation peak value; second, the normalization in Eqs.(6) and (7) helps us to take into account a slight scale difference between the reference and scene images. Computation of the centered and normalized histograms for all circular windows over the reference image as well as the Fourier Transforms can be done as preprocessing. A block diagram of the proposed one-pass matching algorithm is shown in Fig. 3.

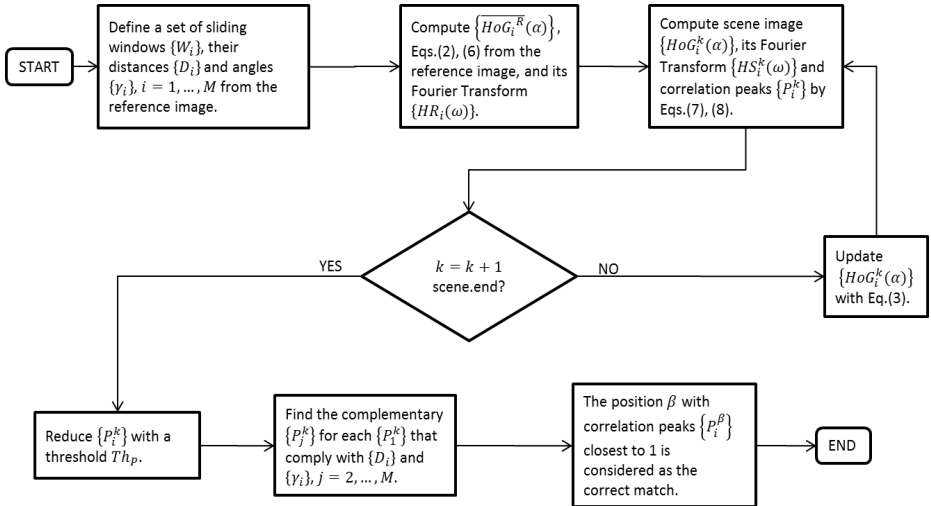


Fig. 3. Block diagram of the algorithm

In order to accelerate the proposed algorithm a decomposition technique [14] is exploited. As shown in Fig 4, an image can be decomposed in a set of small images by decimating in each direction depending on a decomposition level  $L$ ; i. e. when  $L = 2$ , the image is divided in 4 similar images that can reconstruct the original image by performing the inverse process, in a similar way, if  $L = 3$  there are 9 images, and so on.

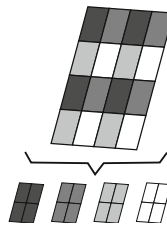
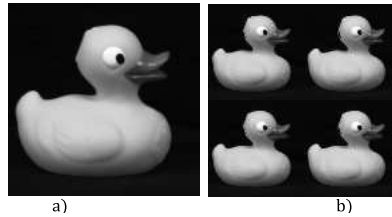


Fig. 4. Illustration of a pyramidal decomposition with level 2 of an image

After selecting the decomposition level, two sets of small images from a scene and reference images are formed. Next, the described algorithm in the preceding section is applied only to one pair of the images. A set of decomposed images are shown in Fig. 5.

The speed-up is achieved by the spatial search in a smaller scene image.



**Fig. 5.** Level 2 of reference image decomposition: a) original image, b) set of decomposed images

### 3 Experimental Results

In this section we present experimental results using the image database ALOI [15]. Ten scene images with the size of  $1280 \times 1024$  pixels and ten reference images of the size  $144 \times 144$  pixels with varied objects were used. Each reference object is located at 100 random positions within the scene images. The performance of the proposed algorithm was compared with that of popular SIFT and SURF algorithms, the parameters of such algorithms are the ones proposed by Lowe [1] and Bay [2]. The algorithms are tested in different conditions such as in-plane/out-of-plane rotations and a slight scaling. The performance is evaluated in terms of the number of correct matches and processing time. The proposed algorithm referred to as CWMA uses two circular windows in each object with a radius  $r$  (depends on the size of object). For a better matching we use  $Q = 64$  bins instead of 9 bins as proposed in [12]. The parameters of the algorithms are as follows:  $M = 2$ ,  $Q = 64$ ,  $Th_p = 0.7$ ,  $L = \{1, 2, 4\}$ . The performance of the tested algorithms for in-plane/out-of-plane rotation is shown in Fig. 6. It can be seen that the CWMA yields the best in-plane rotation invariance and a similar performance with that of the SIFT for out-of-plane rotation. The number in parenthesis is the decomposition level.

Figure 7 illustrates tolerance of the proposed algorithm to image scaling in the range of  $[0.8, 1.2]$ . One may observe that the performance of the CWMA is pretty good for slight image upscaling and downscaling. So, the algorithm can be utilized in real-life application such as tracking.

Finally, the performance of the tested algorithms in terms of processing time is shown in Fig. 8. A standard PC with an Intel Core i7 processor with 3.2 GHz and 8 GB of RAM was used. The implementation of the SIFT and the SURF is taken from the open library OpenCV with Intel multithreading library TBB. The proposed algorithm was also implemented with OpenCV with multithreading from OpenMP library. We see that the proposed algorithm with a pyramidal decomposition performs close to the SURF and outperforms the SIFT.

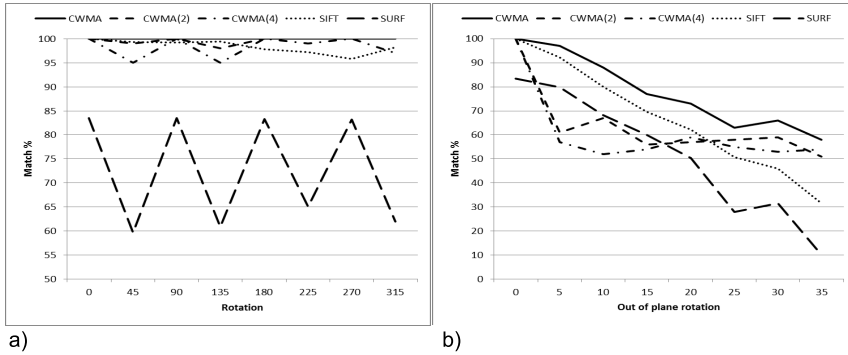


Fig. 6. Matching results for: a) in-plane rotation, b) out-of-plane rotation

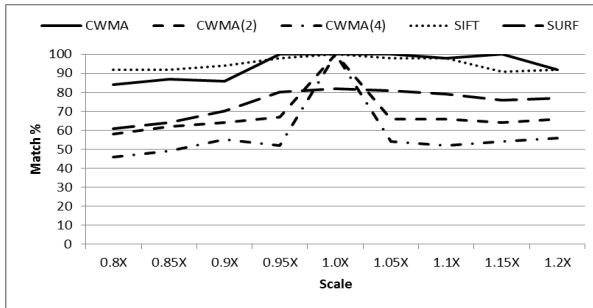


Fig. 7. Matching results for a slight scaling

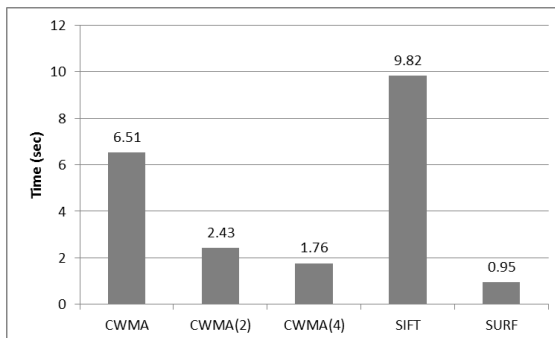


Fig. 8. Processing time of tested algorithms

## 4 Conclusion

In this paper a fast image matching algorithm based on recursive calculation of oriented gradient histograms over several circular sliding windows was presented. Multicore processors with inherent parallel architectures can help to implement the algorithm for image matching with large scenes at high rate. Experimental results showed that the proposed algorithm outperforms the common algorithms for in-plane rotation, yields a similar performance with the SIFT for out-of-plane rotation and a slight scaling, and requires processing time close to the SURF. The algorithm is attractive for real-time applications when rotation invariance matching with a slight scaling is required.

## References

1. Lowe, D.G.: Object recognition from local scale-invariant features. In: Proc. Int. Conference on Computer Vision, vol. 2, pp. 1150–1157 (1999)
2. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: SURF: Speeded Up Robust Features. *Comput. Vis. Image Underst.* 110(3), 346–359 (2008)
3. Calonder, M., Lepetit, V., Strecha, C., Fua, P.: BRIEF: Binary robust independent elementary features. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part IV. LNCS, vol. 6314, pp. 778–792. Springer, Heidelberg (2010)
4. Ortiz, R.: FREAK: Fast Retina Keypoint. In: Proc. of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2012, pp. 510–517 (2012)
5. Liao, C., Wang, G., Miao, Q., Wang, Z., Shi, C., Lin, X.: DSP-Based Parallel Implementation of Speeded-Up Robust Features. *IEICE Trans. on Information and Systems* E94-D(4), 930–933 (2011)
6. Manzurv, T., Zeller, J., Serati, S.: Optical correlator based target detection, recognition, classification, and tracking. *Appl. Opt.* 51, 4976–4983 (2012)
7. Ouerhani, Y., Jridi, M., Alfalou, A., Brosseau, C.: Optimized preprocessing input plane GPU implementation of an optical face recognition technique using a segmented phase only composite filter. *Opt. Comm.* 289, 33–44 (2013)
8. Diaz-Ramirez, V.H., Kober, V.: Adaptive phase-input joint transform correlator. *Appl. Opt.* 46(26), 6543–6551 (2007)
9. Sanders, J., Kandrot, E.: *CUDA by Example: An Introduction to General-Purpose GPU Programming*. Addison-Wesley Professional (2010)
10. Rice, K.L., Taha, T.M., Chowdhury, A.M., Awwal, A.A.S., Woodard, D.L.: Design and acceleration of phase-only filter-based optical pattern recognition for fingerprint identification. *Optical Engineering* 48(11), 117–206 (2009)
11. Zalesky, B.A., Lukashevich, P.V.: Scale Invariant Algorithm to Match Regions on Aero or Satellite Images. In: Proc. Pattern Recognition and Information Processing, vol. 11, pp. 25–30 (2011)
12. Dalal, N., Triggs, B.: Histograms of Oriented Gradients for Human Detection. *Computer Vision and Pattern Recognition* 1, 886–893 (2005)
13. Pratt, W.K.: *Digital Image Processing*. John Wiley & Sons (2007)
14. López-Martínez, J.L., Kober, V.: Fast image restoration algorithm based on camera microscanning. In: Proc. SPIE, vol. 7443, pp. 744310–744315 (2009)
15. Geusebroek, J.M., Burghouts, G.J., Smeulders, A.W.M.: The Amsterdam library of object images. *Int. J. Computer Vision.* 61(1), 103–112 (2005), <http://staff.science.uva.nl/~aloi/>