Efficient NCC-Based Image Matching in Walsh-Hadamard Domain

Wei-Hau Pan, Shou-Der Wei, and Shang-Hong Lai

Department of Computer Science, National Tsing Hua University, Hsinchu, Taiwan
{whpan,greco,lai}@cs.nthu.edu.tw

Abstract. In this paper, we proposed a fast image matching algorithm based on the normalized cross correlation (NCC) by applying the winner-update strategy on the Walsh-Hadamard transform. Walsh-Hadamard transform is an orthogonal transformation that is easy to compute and has nice energy packing capability. Based on the Cauchy-Schwarz inequality, we derive a novel upper bound for the cross-correlation of image matching in the Walsh-Hadamard domain. Applying this upper bound with the winner update search strategy can skip unnecessary calculation, thus significantly reducing the computational burden of NCC-based pattern matching. Experimental results show the proposed algorithm is very efficient for NCC-based image matching under different lighting conditions and noise levels.

Keywords: pattern matching, image matching, image alignment, normalized cross correlation, winner update.

1 Introduction

Pattern matching is widely used in many applications related to computer vision and image processing, such as stereo matching, object tracking, object detection, pattern recognition and video compression, etc. The pattern matching problem can be formulated as follows: Given a source image I and a template image T of size N-by-N, find the best match of template T from the source image I based on the criterion of minimal distortion or maximal correlation. The most popular similarity measures are the sum of absolute differences (SAD), the sum of squared differences (SSD) and the normalized cross correlation (NCC).

For some applications, such as the block motion estimation in video compression, the SAD and SSD measures have been widely used. For practical applications, a number of approximate block matching methods [1][2][3] and some optimal block matching methods [4][5][6] have been proposed. The optimal block matching methods can provide the same solution as that of full search but with less operation by using some early termination schemes in the computation of SAD.

A coarse-to-fine pruning algorithm with the pruning threshold determined from the lower resolution search space was presented in [7]. This search algorithm was proved to provide the global solution with considerable reduction in computational cost. Hel-Or and Hel-Or [8] proposed a fast template matching method based on accumulating the distortion on the Walsh-Hadamard domain in the order of the frequency of the

D. Forsyth, P. Torr, and A. Zisserman (Eds.): ECCV 2008, Part III, LNCS 5304, pp. 468–480, 2008.

Walsh-Hadamard basis. By using a predefined threshold, they can reject early most of the impossible candidates very efficient. However, this algorithm can only be applied for the SSD measure, and it is not guaranteed to find the globally optimal solution. More recently, Pele and Werman [18] proposed a very fast method for pattern matching based on Bayesian sequential hypothesis testing. They used a rank to decide how many pixels can be skipped during the sliding window search process. Similarly, this method cannot guarantee to find the globally optimal solution.

Chen et al. [9] proposed a winner-update algorithm for fast block matching based on the SAD and SSD measures. This algorithm significantly reduces the computation and guarantees to find the globally optimal solution. In their algorithm, only the current winner location with the minimal accumulated distortion is considered for updating the accumulated distortion. This updating process is repeated until the winner has gone through all levels in the pyramids that are constructed from the template and the candidate windows for the distortion calculation. The winner update algorithm examines all the candidates in the search image to guarantee the global optimal solution but skips the unnecessary calculations of the distortion measures for most candidates.

The NCC similarity measure is commonly used for pattern matching under different lighting conditions. The definition of NCC is given as follows:

$$NCC(x,y)=\frac{\sum_{i=1}^{N}\sum_{j=1}^{N} I(x+i,y+j)\cdot T(i,j)}{\sqrt{\sum_{i=1}^{N}\sum_{j=1}^{N} I(x+i,y+j)^2}\cdot\sqrt{\sum_{i=1}^{N}\sum_{j=1}^{N} T(i,j)^2}} \tag{1}$$

The NCC measure is more robust than SAD and SSD under uniform illumination changes, so the NCC measure has been widely used in object recognition and industrial inspection. The correlation approach is very popular for image registration [15]. The traditional NCC needs to compute the numerator and denominator at all locations in the image, which is very time-consuming. Lewis [12] employed the sum table scheme [13][14] to reduce the computation in the denominator. After building the sum table for the source image, the block squared intensity sum for any window inside the source image can be calculated very efficiently with 4 simple operations. Although the sum table scheme can significantly reduce the computation of the denominator in the NCC-based pattern matching, it is strongly demanded to simplify the computation involved in the numerator of the NCC-based pattern matching. The FFT-based method has been employed to calculate the cross correlation all over the image via element-wise multiplication in the frequency domain [16]. It is very effective especially when the size of the template is comparable to the size of the search image. Stefano and Mattoccia [10][11] derived upper bounds for the cross correlation based on Jensen's and Cauchy-Schwarz inequalities to early terminate some search points. They partitioned the search window into two blocks and compute the partial cross correlation for the first block with the other blocks bounded by the upper bound. Then, the successive elimination algorithm (SEA) was applied to reject the impossible candidates successively. More recently, Mahmood and Khan [17] proposed new tighter bounds on correlation for fast block matching in video compression. They derived tighter bounds for the B-frames in video by using the correlation between the reference frame and the C-frame.

In this paper, we propose an efficient NCC-based pattern search algorithm by applying the winner update procedure on the Walsh-Hadamard domain. The winner update scheme is employed with the upper boundary value for the NCC derived from the Cauchy-Schwarz inequality in the Walsh-Hadamard basis to skip unnecessary calculation. The rest of this paper is organized as follow: we first briefly review the winner update scheme and the SEA-based method for optimal pattern matching [4][5] as well as the upper bound for the cross correlation derived from the Cauchy-Schwarz inequality [10][11]. Then, we present the proposed efficient NCC-based image matching algorithm that performs the winner update scheme in the Walsh-Hadamard domain in section 3. The implementation details and the experimental results are shown in section 4 and section 5, respectively. Finally, we conclude this paper in the last section.

2 Previous Works

Although the NCC measure is more robust than SAD, the computational cost of NCC is very high. The technique of the sum table [12][13][14] can be used to reduce the computation involved in the denominator in NCC. Any block sum in the source image can be accomplished with 4 simple operations from the pre-computed sum table. To reduce the computational cost in the numerator, Stefano and Mattoccia [10][11] derived upper bounds for the cross correlation based on Jensen's and Cauchy-Schwarz inequalities to terminate early some impossible search points. Because the bound is not very tight, they partitioned the search window into two blocks and compute the partial cross correlation for the first block (from row 1 to row k) with the other blocks bounded by the upper bound (from row k to row N). Then they used the SEA scheme to reject the impossible candidates successively. Based on the Cauchy-Schwarz inequality [11] given below

$$\sum_{i=1}^{N} a_i \cdot b_i \leq \sqrt{\sum_{i=1}^{N} a_i^2} \cdot \sqrt{\sum_{i=1}^{N} b_i^2} \tag{2}$$

the upper bound (UB) of the numerator, i.e. cross correlation, can be derived as follows:

$$\begin{aligned} UB(x,y) &= \sum_{i=1}^{M}\sum_{j=1}^{k} I(x+i,y+j)\cdot T(i,j) \\ &+ \sqrt{\sum_{i=1}^{M}\sum_{j=k+1}^{N} I(x+i,y+j)^2} \cdot \sqrt{\sum_{i=1}^{M}\sum_{j=k+1}^{N} T(i,j)^2} \\ &\geq \sum_{i=1}^{M}\sum_{j=1}^{N} I(x+i,y+j)\cdot T(i,j) \end{aligned} \tag{3}$$

Thus, the boundary value of NCC is given by:

$$BV(x,y) = \frac{UB(x,y)}{\left(\sum_{(i,j)\in W} I^2(x+i,y+j)\right)^{\frac{1}{2}} \cdot \left(\sum_{(i,j)\in W} T^2(i,j)\right)^{\frac{1}{2}}} \tag{4}$$

where W is the window for the template. Similar to the SEA scheme, the candidate at the position (x,y) of image I is rejected if $BV(x, y) < NCC_{\max}$, and $NCC_{\max}$ is updated by $NCC(x, y)$ if $NCC(x, y) > NCC_{\max}$.

3 The Proposed Efficient NCC-Based Image Matching Algorithm

Hel-Or and Hel-Or [1] proposed an efficient algorithm to eliminate the impossible candidates for SSD-based pattern search when the accumulated distortions in the Walsh-Hadamard domain exceed a predefined threshold. The elimination is efficient because the first few lowest-frequency Walsh-Hadamard coefficients usually occupy most of the energy in the SSD. With an appropriate threshold, they can reject most of the impossible candidates in a very early stage, thus leading to a very efficient algorithm. However, the selection of an appropriate threshold is very critical to the performance of the algorithm. In addition, it is not clear how to extend their algorithm for NCC-based image matching.

In this section, we derive a novel boundary value for NCC in a projection domain corresponding to an orthogonal transformation by using the Cauchy-Schwarz inequality. Because of its nice energy packing property and ease of computation, we select the Walsh-Hadamard transform as the projection kernel for efficient pattern matching. In addition, we combine the winner update strategy with the hierarchical order of the boundary values for the Walsh-Hadamard domain to efficiently find the best match with the NCC criterion from the source image.

3.1 The Novel Boundary Value of Numerator in NCC Using Orthonormal Projections and Cauchy-Schwarz Inequality

The cross correlation values between two vectors (images) is the same as the cross correlation between their transformed vectors when an orthogonal transformation is employed. This statement can be easily proved in the following. Let u and v be two n-dimensional column vectors. Consider an orthogonal transformation matrix $\mathbf{P} \in R^{n \times n}$ is applied to transform **u** and **v**, and **a** and **b** be the corresponding transformed vectors, respectively, i.e. $a = Pu$ and $b = Pv$. Thus, we have

$$a \cdot b = a^T b = (Pu)^T (Pv) = u^T P^T Pv = u^T v = u \cdot v \tag{5}$$

Similarly, we can easily prove that the 2-norms of the original vector and its transformed vector are the same, i.e. $\|\mathbf{a}\| = \|\mathbf{u}\|$ and $\|\mathbf{b}\| = \|\mathbf{v}\|$. It is obvious that the numerator and the denominator in NCC are invariant under an orthogonal transformation.

For reducing the computation burden of the numerator in NCC, we derive an upper bound for the numerator by using the Cauchy-Schwarz inequality on the projection domain with an orthnormal basis.

We can rewrite the cross correlation (CC) in the numerator of NCC by partitioning the summation of the components in the vectors **a** and **b** into two parts as below:

$$\sum_{i=1}^{N} a_i \cdot b_i = \sum_{i=1}^{k} a_i \cdot b_i + \sum_{i=k+1}^{N} a_i \cdot b_i \tag{6}$$

The second term of right hand side in equation (6) can be bounded by Cauchy-Schwarz inequality as below:

$$\begin{aligned} \sum_{i=k+1}^{N} a_i \cdot b_i &\le \sqrt{\sum_{i=k+1}^{N} a_i^2} \cdot \sqrt{\sum_{i=k+1}^{N} b_i^2} \\ &= \sqrt{(\sum_{i=1}^{N} a_i^2 - \sum_{i=1}^{k} a_i^2)} \cdot \sqrt{\sum_{i=k+1}^{N} b_i^2} \end{aligned} \tag{7}$$

Thus, we have an upper bound for the numerator in NCC as follows:

$$\sum_{i=1}^{N} a_i \cdot b_i \le \sum_{i=1}^{k} a_i \cdot b_i + \sqrt{(\sum_{i=1}^{N} a_i^2 - \sum_{i=1}^{k} a_i^2) \cdot \sum_{i=k+1}^{N} b_i^2} = UB_k \tag{8}$$

The second term of the upper bound in equation (8) is bounded by the Cauchy-Schwarz inequality, so we have the hierarchical order for the upper bounds of the cross correlation in NCC as follows:

$$UB_1 \ge UB_2 \ge .... \ge UB_N = \sum_{i=1}^{N} a_i \cdot b_i \tag{9}$$

Then we can define the boundary value (BV) of NCC at the *k-th* level by

$$BV_k = \frac{UB_k}{\sqrt{\sum_{i=1}^{N} a_i^2} \cdot \sqrt{\sum_{i=1}^{N} b_i^2}} \tag{10}$$

From equation (9) and (10), we can obtain the hierarchical order of the boundary values for NCC at different levels as follows:

$$1 = BV_0 \ge BV_1 \ge .... \ge BV_N = NCC \tag{11}$$

The terms $\sum_{i=1}^{N} a_i^2$ and $\sum_{i=1}^{N} b_i^2$ in the denominator of equation (10) and in equation (8) can be calculated very efficiently in the original domain by using the integral image. Thus, the calculation of *BV* becomes very efficient, since we only need to update the accumulated sums $\sum_{i=1}^{k} a_i \cdot b_i$ and $\sum_{i=1}^{k} a_i^2$ at different levels.

3.2 The Proposed Method by Combining the Novel Boundary Value with the Winner Update Strategy

With the nice energy-packing property and the ease of computation, we use the Walsh-Hadamard transform as our projection kernel in the order of the frequency of the Walsh-Hadamard basis and use the Walsh-Hadamard tree [1] to accelerate the computation of WH(Walsh-Hadamard) coefficients. The vectors **a** and **b** in equation (5) can be the transformed vectors of the candidate and template in the Walsh-Hadamard domain. We applied the winner update strategy on the novel hierarchical order of the boundary values at different levels as described in section 3.1 to find the best match in the source image.

The winner update scheme choose the candidate with the largest boundary value to be the best match as the temporary winner from the candidate pool. The temporary winner is iteratively selected and updated until one has accumulated the boundary value of all WH coefficients. Similar to Chen et al. [9], we also use a hash table to find the temporary winner very efficiently.

In the beginning, we compute the square sum of the template, denoted by *Tss*, and the windowed square sum for all candidates in the search image, denoted by *Iss*, as follows:

$$Tss = \sum_{i=1}^{N}\sum_{j=1}^{N} T^2(i,j) \tag{12}$$

$$Iss(x,y) = \sum_{i=1}^{N}\sum_{j=1}^{N} I^2(x+i, y+j) \tag{13}$$

Note that the windowed square sum *Iss* can be easily computed with an integral square image. We first calculate all the WH coefficients of the template as vector **b** and the first WH coefficient (DC term) of all the candidates in the source image. We can calculate BV_1 from the first WH coefficient and build the hash table to find temporary winner.

At each iteration, we find the candidate with the largest BV as the temporary winner from hash table and update the boundary value and the associated level. We compute the next WH coefficient of the temporary winner, denoted by a_k, and calculate the upper bound UB_k with the following equation.

$$UB_k = \sum_{i=1}^{k} a_i \cdot b_i + \sqrt{(Iss - \sum_{i=1}^{k} a_i^{\,2}) \cdot (Tss - \sum_{i=1}^{k} b_i^{\,2})} \tag{14}$$

Note that the terms Iss, Tss and the projected template, denoted as vector **b**, in equation (14) have been calculated at the first step. At each step, we only need to update the accumulated sums $\sum_{i=1}^{k} a_i^{\,2} = \sum_{i=1}^{k-1} a_i^{\,2} + a_k^{\,2}$, $\sum_{i=1}^{k} b_i^{\,2} = \sum_{i=1}^{k-1} b_i^{\,2} + b_k^{\,2}$ and $\sum_{i=1}^{k} a_i \cdot b_i = \sum_{i=1}^{k-1} a_i \cdot b_i + a_k \cdot b_k \cdot$

Thus, we can obtain the new BV at the next level from equation (10). After updating the boundary value and the level, we push the temporary winner into the hash table. This winner update procedure is repeated until one candidate has reached the final level.

Algorithm 2. The proposed fast NCC-based pattern matching algorithm

1. Initialization

1.1: Build the integral square image for the source image

1.2: Calculate the square sum of template *Tss*.

1.3: Calculate the window square sum *Iss(x,y)* of all candidates from the integral square image.

1.4: Calculate the projected vector b of the template.

1.5: Calculate the BV_1 for all candidates and initialize the Hash Table.

2. Winner Update Scheme

Repeat

2.1: Select the candidate *I(x,y)* with maximal *BV* in hash table as the temporary winner

2.2: Update the level *l* and the *BV* of the temporary winner

2.2.1. Compute the next WH coefficient a_{l+1} of the temporary winner. Update its level $l=l+1$;

2.2.2. Calculate UB_l for level *l*. Compute $BV_1 = UB_l \big/ \sqrt{Tss \cdot Iss(x,y)}$

2.2.3. Push the temporary winner into the Hash Table.

Until the winner reaches the maximal level.

The proposed fast algorithm can be easily extended to the zero-mean normalized cross correlation (ZNCC) by rewriting it in the following form:

$$\frac{\sum_{i=1}^{M}\sum_{j=1}^{N}(I(x+i,y+j)-\overline{I}(x,y))\cdot(T(i,j)-\overline{T})}{\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}(I(x+i,y+j)-\overline{I}(x,y))^2}\cdot\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}(T(i,j)-\overline{T})^2}}$$
$$=\frac{\sum_{i=1}^{M}\sum_{j=1}^{N}I(x+i,y+j)\cdot T(i,j)-MN\overline{I}(x,y)\overline{T})}{\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}I^2(x+i,y+j)-MN\overline{I}^2(x,y)}\cdot\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}T^2(i,j)-MN\overline{T}^2}} \tag{15}$$

where

$$\overline{I}(x,y)=\frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}I(x+i,y+j)\text{, }\overline{T}=\sum_{i=1}^{M}\sum_{j=1}^{N}T(i,j) \tag{16}$$

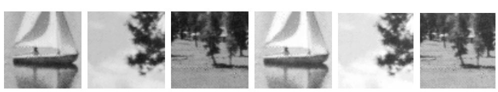
Note that the summations of the image intensities and the squared intensities in the local window at each location in the image can be computed very efficiently by using the corresponding integral images.

4 Experimental Results

In this section, we demonstrate the efficiency of the proposed NCC-based pattern matching algorithm in the Walsh-Hadamard domain. The proposed algorithm incrementally calculates higher level Walsh-Hadamard coefficients of the best candidate to obtain successively tighter upper bounds for the normalized cross correlation. To compare the efficiency of the proposed algorithm, we also implemented the BPC [11] method with the correlation ratio Cr=50% and Hel-Or & Hel-Or's method [8] (with default parameter RejectThreshold set to 10). The experiments were performed on a PC with an Intel Pentium M 1.73GHz CPU and 512MB RAM. In the first experiment, we used the sailboat image of size 512-by-512 and its noisy version as the source images as shown in Figure 1. Six template images of size 64x64 were selected from the original sailboat image as depicted in Figure 2. The templates in Figure 2(d), (e), and (f) are the brighter version (increased by 30 intensity grayscales for all pixels) of the original templates in Figure 2(a), (b), and (c), respectively. To compare the robustness and the efficiency of the proposed algorithms, we add random Gaussian noises with variance 10 onto the search image as shown in Figure 1(b) and compare the performance of different pattern matching methods on the noisy image. The full search, BPC and the proposed algorithm are guaranteed to find the optimal NCC solution from the search image, so we only focus on the comparison of search time required for these three algorithms. Note that all the three methods used the sum table to reduce the computation in the denominator of NCC. In addition, the pattern search method by Hel-Or and Hel-Or [8] is not guaranteed to find a globally optimal solution. The execution time required for the full search, BPC, Hel-Or & Hel-Or's method and the proposed algorithm are shown in Table 1 and 2. For a fair comparison, the execution time shown here includes the time required for memory allocation for sum table and Walsh-Hadamard transform. We also conducted the same experiments for the airplane image. Table 3 and 4 summarize the experimental results. All of these experimental results show the proposed NCC-based pattern matching algorithm in Walsh-Hadamard domain significantly outperforms the full search and BPC methods. In addition, the computational efficiency of the proposed algorithm is similar to that of Hel-Or and Hel-Or's method, but their method failed to find the optimal solution in several pattern matching experiments, especially when there is intensity scaling in the template.

(a) (b)

Fig. 1. (a) The original "sailboat" image and (b) the noisy sailboat image added with zero-mean Gaussian noise with variance 10



(a) (b) (c) (d) (e) (f)

Fig. 2. (a), (b) & (c) are three template images of size 64-by-64, and (d), (e) & (f) are their corresponding brighter versions (increased by 30 intensity grayscales for all pixels)

Table 1. The execution time (in milliseconds) of applying the full-search (NCC), BPC, Hel-Or and Hel-Or's method and the proposed algorithm to the NCC-based pattern matching with the six templates shown in Figure 2(a)~(f) and the source image shown in Figure 1(a)

ms	T(a)	T(b)	T(c)	T(d)	T(e)	T(f)
Full-Search (NCC)	3163	3162	3180	3171	3162	3198
BPC	1590	1812	1784	1597	1811	1752
Hel-Or & Hel-Or [8]	61	67	70	*74	*52	*54
Proposed Algorithm	66	68	70	70	72	80

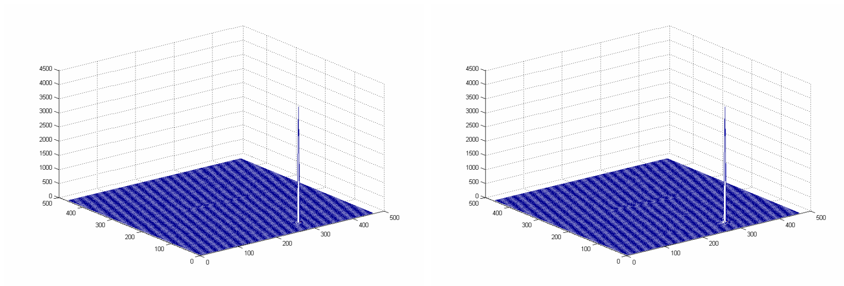
Table 2. The execution time (in milliseconds) of applying the full-search (NCC), BPC, Hel-Or and Hel-Or's method and the proposed algorithm to the NCC-based pattern matching with the six templates shown in Figure 2(a)~(f) and the source image shown in Figure 1(b)

ms	T(a)	T(b)	T(c)	T(d)	T(e)	T(f)
Full-Search (NCC)	3180	3173	3179	3172	3173	3177
BPC	1625	1806	1817	1614	1801	1755
Hel-Or & Hel-Or's [8]	59	*73	72	*76	*52	*55
Proposed Algorithm	80	95	99	85	110	117

* indicates that method finds an incorrect matching result.

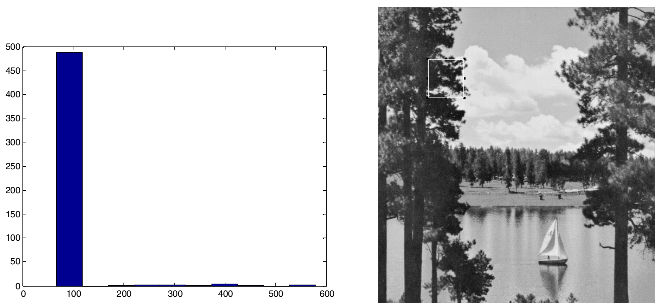
To show the influence of noise corruption on the efficiency of winner update in the proposed algorithm, we show in Figure 3(a) and 3(b) the total numbers of winner updates at all locations in the search image to find the template shown in Figure 2(a) from the original sailboat image and its noisy version shown in Figure 1(a) and Figure 1(b), respectively. The total number of candidates is 201601 (449x449) and the average winner update counts are 1.1506 and 1.2775 for the clean and noisy images, respectively. Note that the peaks of winner update counts shown in the figures correspond to the final searched locations. It is evident that the proposed algorithm is very efficient since all the locations other than the peak solution have very small numbers of winner updates. We can also see that the winner updates in the noisy search image as shown in Figure 3(b) are slightly more than those in the clean image as shown in Figure 3(a).

To show the performance of the proposed algorithm for template matching tasks, we randomly selected 500 different templates of sizes 64-by-64 from the sailboat and airplane images with enough gradient magnitude to avoid selecting homogeneous blocks. The histograms of the execution time are depicted in Figure 4(a). From these figures, we can see the proposed method takes about 60-90 milliseconds to find most

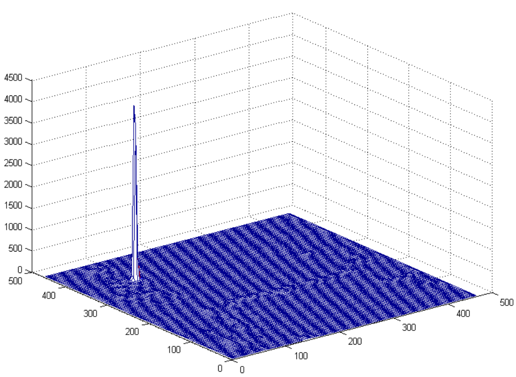


(a) (b)

Fig. 3. The total numbers of winner update at all locations in the search image by applying the proposed method to find the template in Figure 2(a) from the source sailboat image with (a) no noise (Figure 1(a)) and (b) additive Gaussian noises (Figure 1(b))



(a) (b)



(c)

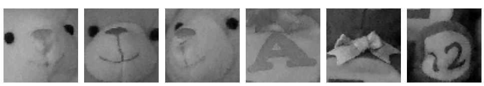
Fig. 4. (a) The histograms of the execution time (in milliseconds) required for applying the proposed algorithm to find 500 randomly selected templates of size 64-by-64 from the sailboat image. (b) The inefficient template matching block randomly selected from the sailboat image, and (c) the winner update numbers of applying the proposed algorithm to find this template from the original image.

of these randomly selected templates. The execution time required for the proposed template matching algorithm depends on the image content in the template and the search images. The more unique content in the template, the faster the proposed algorithm finds the template from the image.

In Figure 4(a), there are some special cases with execution time more than 300 milliseconds. For one of these cases, the template in the sailboat image is shown in Figure 4(b), and the winner update numbers of applying the proposed algorithm on this template are depicted in Figure 4(c). The reason that causes the inefficient matching is the candidates in the neighborhood of best match have similar Walsh-Hadamard coefficients. It is clear from Figure 4(c) that there are many updating numbers in the neighborhood of the best match.

(a) (b)

Fig. 5. (a) The original 700x700 "Bears" image and (b) the darker 700x700 "Bears" image



(a) (b) (c) (d) (e) (f)

Fig. 6. Testing 64x64 templates cut from Figure 5(b)

Table 3. The execution time (in milliseconds) of applying the full-search (NCC), BPC, Hel-Or & Hel-Or's method and the proposed algorithm to the NCC-based pattern matching with the six templates shown in Figure 6(a)~(f) and the source image shown in Figure 5(a)

ms	T(a)	T(b)	T(c)	T(d)	T(e)	T(f)
Full-Search (NCC)	6514	6520	6511	6525	6531	6522
BPC	4266	4281	4293	3913	4337	4396
Hel-Or & Hel-Or's [8]	*158	*156	*155	*172	*125	*140
Proposed Algorithm	196	420	228	233	211	292

* indicates that method finds an incorrect matching result.

In addition, we applied the proposed fast NCC-based pattern matching algorithm to real source and template images acquired under different illumination conditions. Figure 5(a) and Figure 5(b) are two photographs taken from different lighting illumination conditions. We cut some 64x64 templates from the darker photo, i.e. Figure 5(b), as shown in Figure 6, and apply the different pattern matching methods to find these templates from the source image shown in Figure 5(a). Table 3 summarizes the experiment results.

5 Conclusion

In this paper, we proposed a very efficient algorithm for NCC-based pattern search in the Walsh-Hadamard domain. We derived a novel upper bound for cross correlation in NCC in an orthogonal transform domain. To achieve very efficient computation, we used Walsh-Hadamard transformation as the projection kernel. For the NCC pattern search, the winner update scheme is applied in conjunction with the novel incremental upper bound for the cross correlation derived from Cauchy-Schwarz inequality. The experimental results show the proposed algorithm is very efficient and robust for pattern matching under different illumination changes and noisy environments.

References

1. Zhu, S., Ma, K.K.: A new diamond search algorithm for fast block matching motion estimation. IEEE Trans. Image Processing 9(2), 287–290 (2000)
2. Li, R., Zeng, B., Liou, M.L.: A new three-step search algorithm for block motion estimation. IEEE Trans. Circuits Systems Video Technology 4(4), 438–442 (1994)
3. Po, L.M., Ma, W.C.: A novel four-step search algorithm for fast block motion estimation. IEEE Trans. Circuits Syst. Video Technol. 6, 313–317 (1996)
4. Li, W., Salari, E.: Successive elimination algorithm for motion estimation. IEEE Trans. Image Processing 4(1), 105–107 (1995)
5. Gao, X.Q., Duanmu, C.J., Zou, C.R.: A multilevel successive elimination algorithm for block matching motion estimation. IEEE Trans. Image Processing 9(3), 501–504 (2000)
6. Lee, C.-H., Chen, L.-H.: A fast motion estimation algorithm based on the block sum pyramid. IEEE Trans. on Image Processing 6(11), 1587–1591 (1997)
7. Gharavi-Alkhansari, M.: A fast globally optimal algorithm for template matching using low-resolution pruning. IEEE Trans. Image Processing 10(4), 526–533 (2001)
8. Hel-Or, Y., Hel-Or, H.: Real-time pattern matching using projection kernels. IEEE Trans. Pattern Analysis Machine Intelligence 27(9), 1430–1445 (2005)
9. Chen, Y.S., Huang, Y.P., Fuh, C.S.: A fast block matching algorithm based on the winner-update strategy. IEEE Trans. Image Processing 10(8), 1212–1222 (2001)
10. Di Stefano, L., Mattoccia, S.: Fast template matching using bounded partial correlation. Machine Vision and Applications 13(4), 213–221 (2003)
11. Di Stefano, L., Mattoccia, S.: A Sufficient Condition based on the Cauchy-Schwarz Inequality for Efficient Template Matching. In: IEEE International Conf. Image Processing, Barcelona, Spain, September 14-17 (2003)
12. Lewis, J.P.: "Fast template matching," Vision Interface, pp. 120–123 (1995)

13. Mc Donnel, M.: Box-filtering techniques. Computer Graphics and Image Processing 17, 65–70 (1981)
14. Viola, P., Jones, M.: Robust real-time face detection. International Journal of Computer Vision 52(2), 137–154 (2004)
15. Zitová, B., Flusser, J.: Image registration methods: a survey. Image Vision Computing 21(11), 977–1000 (2003)
16. Brown, L.G.: A survey of image registration techniques. ACM Computing Surveys 24(4), 325–376 (1992)
17. Mahmood, A., Kahn, S.: Exploiting Inter-frame Correlation for Fast Video to Reference Image Alignment. In: Proc. 8th Asian Conference on Computer Vision (2007)
18. Pele, O., Werman, M.: Robust real time pattern matching using Bayesian sequential hypothesis testing. IEEE Trans. Pattern Analysis Machine Intelligence (to appear)