FAST TEMPLATE MATCHING BY APPLYING WINNER-UPDATE ON WALSH-HADAMARD DOMAIN

Shou-Der Wei, Shao-Wei Liu, Shang-Hong Lai

Department of Computer Science, National Tsing Hua University, Hsinchu, Taiwan

ABSTRACT

Fast template matching is strongly demanded for many practical applications related to computer vision and image processing. In this paper, we propose a fast template matching method by applying the winner-update strategy on the Walsh-Hadamard domain. By taking advantage of the nice energy packing property of the Walsh-Hadamard transformation, we can just apply the winner-update process with a small number of Walsh-Hadamard coefficients to reduce the computational burden for template matching in an image. Experimental results demonstrate the efficiency and robustness of the proposed template matching algorithm under different noise levels.

***Index Terms*—**Walsh-Hadmard, Winner update, Template matching, Pattern matching.

1. INTRODUCTION

Template matching is widely used in many applications related to computer vision and image processing, such as object tracking, object detection, pattern recognition and the video compression, etc. The template matching problem can be formulated as follows: Given a source image I of size MxN and a template image T of size kxk, the goal of template matching is to find the best match of template T from the source image I. For defining the best match, the sum of square difference (SSD) is the commonly used distortion measure given as follows:

$$D(u,v)=\sum_{i=0}^{k}\sum_{j=0}^{k}(T(i,j)-I(u+i,v+j))^2 \tag{1}$$

The exhaustive search method can be used to find the best match but it is very time-consuming. For practical applications, a fast template matching method is strongly demanded. For video compression, a number of approximate block matching methods have been proposed [4][5][6][7]. In [3], a coarse-to-fine pruning algorithm with the pruning threshold determined from the lower resolution search space was presented. This search algorithm can be proved to provide the global solution.

Recently, Hel-Or and Hel-Or [1] proposed a fast template matching method based on accumulating the distortion on the Walsh-Hadamard domain in the order of the frequency of the Walsh-Hadamard basis. In general, a small number of the first few projections can capture most of the distortion energy. By using a predefined threshold, they can early reject most of the impossible candidates very efficient. A Walsh-Hadamard tree structure was also employed in their work to efficiently calculate the Walsh-Hadamard coefficients.

Chen et al. [2] proposed a fast block matching algorithm based on the winner-update strategy, which can reduce a large portion of computation and guarantee 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 the levels of 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.

In the work by Hel-Or and Hel-Or [1], they used a predefined threshold to reject the impossible candidates during image comparison in the Walsh-Hadamard domain. However, it is very difficult to determine appropriate thresholds for different templates and source images. Instead of using predefined thresholds, we improve this approach by applying the winner-update scheme on Walsh-Hadamard domain and accumulate the distortion according to the order of frequency in the Walsh-Hadamard domain. The efficiency of the winner update scheme is determined by the total number of updating. Because a few low-frequency Walsh-Hadamard coefficients can capture a large proportion of the difference between the template and the candidate window, we can determine the most possible candidate based on accumulating the first few coefficient deviations and updating them until going through all coefficients to find the final winner.

The rest of this paper is organized as follow: In the next two sections, we briefly describe the Walsh-Hadamard transform and the winner-update scheme. Then, we present the proposed method that performs the winner-update scheme on the Walsh-Hadamard basis in section 4. Some experimental results with comparisons to previous methods are given in section 5. Finally, we conclude this paper in the last section.

2. WALSH-HADAMARD TRANSFORM

Walsh-Hadamard (WH) transform has been widely used in many fields, such as image compression, object detection and pattern matching. Each WH basis vector can be regarded as a Haar-like filter. The WH coefficients can be computed by applying the Haar-like filtering to the image. Since the WH basis vectors are composed of binary elements, computing the corresponding coefficients only requires the addition and subtraction operations. Therefore, its computational cost is very efficient. Because Walsh-Hadamard transform has an orthonormal basis, the distortion measure in the image pixel space is the same as the corresponding distortion in the WH transform domain. Due to the nice energy packing property of the WH transform, we can use a small number of differences in the most dominant WH coefficients to approximate the actual distance between the template and candidates to save the computational cost. This property plays an important role in our fast template matching algorithm, and it makes the winner-update scheme more efficient.

Note that the Walsh-Hadamard coefficients of an image I are computed from the following equations:

$$WH(u,v) = \frac{1}{N}\sum_{r=0}^{N-1}\sum_{c=0}^{N-1} I(r,c)\times(-1)^{\sum_{i=0}^{n-1}[bi(r)pi(u)+bi(c)pi(v)]} \tag{2}$$

$$P_0(u) = b_{n-1}(u)$$

$$P_s(u) = b_{n-s}(u) + b_{n-s-1}(u), \quad where,\ s = 1 \sim n-1 \tag{3}$$

where N = 2^n is the row/column size of the image and $b_i(u)$ is the i^{th} bit of u in the binary representation.

3. WINNER UPDATE STRATEGY

In this section, we use a simple poker games to describe the concept of the winner-update algorithm [2]. As illustrated in Figure 1, suppose there are four players in the poker game and each player has four cards. The player with the minimum total points of his cards is the winner of the poker game. At the first stage, every players show one of their cards, and the card's point is added into the accumulated sum. Subsequently, the current winner, i.e. the player with the minimum accumulated sum, show the next card and add this card's point to update his accumulated sum. The above procedure is repeated iteratively until one player has showed all of his cards and his accumulated sum is the smallest among all, and then this player is the winner of the game. An example of the winner update scheme is illustrated in Figure 1. After the first stage, player 3 is the temporary winner and he shows the next card. After updating the accumulated sum of player 3, player 1 becomes the temporary winner. Following the winner update procedure described above, each player alternatively becomes the temporary winner and finally player 3 is the winner that shows all his cards with the minimal accumulated sum.

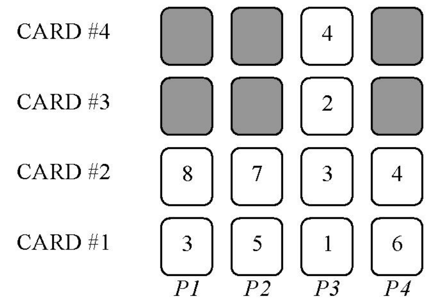


Fig. 1: An illustration of the winner-update strategy with a simple poker game.

4. THE PROPOSED METHOD

Hel-Or and Hel-Or [1] used a predefined threshold to eliminate the impossible candidates for pattern search when the accumulated distortions exceed a predefined threshold. The elimination is efficient because the first few Walsh-Hadamard coefficients usually occupy most of the energy in the SSD. The drawback of their method is the requirement of an appropriate threshold for each pattern search task. This threshold depends on the template, source image and the noise level. Therefore, it is difficult to automatically determine an optimal threshold.

In this paper, we applied the winner-update strategy on the Walsh-Hadamard domain to find the best match position on source image without requiring any predefined threshold. The efficiency of winner-update strategy [2] depends on the total number of winner updates. The winner-update strategy [2] is applied on the Walsh-Hadamard domain because of its nice energy packing property. A small number of differences in the corresponding most dominant Walsh-Hadamard coefficients are used to approximate the actual distance between the template and candidate images. Thus we can reduce the total number of winner updates and save the computational cost.

We also use the Walsh-Hadamard tree [1] to compute the WH coefficients efficiently. In the beginning, we compute all the coefficients of the template and the first WH coefficient (DC term) of all the windows in the source image. Then, we compute the squared differences of the first coefficient between the template and all the windows in the source image and find the temporary winner with the minimal squared difference using the hash table. The next coefficient of the temporary winner is then computed, in order to determine the difference from the template. Summing up the differences between those WH coefficients calculated before makes the lower bound tighter. The temporary winner is iteratively selected and updated until one has accumulated the distortions of all WH coefficients. Then, the SSDs for all the candidates in the same hash table entry are computed to determine the final best matched candidate. The procedures of the proposed Walsh-Hadamard based winner update technique for pattern matching are summarized as follows:

1. Compute all Walsh-Hadamard coefficients of the template and the DC terms of all windows in the source image.
2. Initialize the accumulated distortions for all windows as the squared differences between their corresponding DC coefficients and that of the template, and put them into a Hash table.
3. Select the window with the smallest accumulated distortion from the Hash table.
4. Compute the next Walsh-Hadamard coefficient of the selected window and its difference with that of the template. Update the accumulated distortion of the selected window.
5. Repeat step 3 and 4 until the winner reaches the last coefficients.
6. Compute the SSDs of all the candidates in the same entry of the Hash table and select the window with the minimal SSD to be the final winner.

5. EXPERIMENTAL RESULTS

Figure 2 shows the source image of size 512-by-512 and two template images of size of 32x32 as indicated by the rectangles. The rectangle on the left eye is template 1, and the rectangle on the lip position is template 2. The results of applying the proposed method and the projection-kernel based pattern matching method [1] on the problem given in Figure 2 are shown in Table 1. The execution time shown in Table 1 includes the time of pattern matching memory allocation for Walsh-Hadamard tree and the Hash table, and building WH tree. When the thresholds are below 20 and 15, the results of applying the projection kernel based method [1] to template 1 and template 2, respectively, are incorrect, so we set the relative grid to be blank. Both methods outperform the full search. Although the projection kernel based method with the optimal threshold is more efficient than the proposed method, it is difficult to determine the optimal threshold for each different image in practice. If the predefined threshold is too small, the projection kernel based method may not find the pattern. If the threshold is too large, it becomes very time-consuming.

To compare the robustness and the efficiency of the proposed algorithm and the projection kernel based method [1], we add random Gaussian noises of different levels onto the search image to perform the pattern search. Figure 3(a) and (b) show the source image added with random Gaussian noises with standard deviations of 18 and 31, respectively. The results of applying both methods for this experiment are listed in Figure 4. The value d in Figure 4 denotes the standard deviation of Gaussian noises. We can see from Figure 4 that larger noises lead to longer executing time. It is also evident from this experiment that the optimal thresholds for the projection kernel based method are quite different for different noise levels, while our algorithm does not require such threshold selection.

Fig. 2: The source image and the two templates.

Table 1: The execution time of the full search, the proposed method and the projection kernel based method with different thresholds.

Threshold	10	15	20	25	30	35	40	45	50
T1 Hel-Or's	-	-	78	94	78	109	140	219	234
T1 Proposed	125	125	125	125	125	125	125	125	125
T2 Hel-Or's	-	78	49	125	219	375	1735	8031	9329
T2 Proposed	172	172	172	172	172	172	172	172	172
FS	5344	5344	5344	5344	5344	5344	5344	5344	5344

6. CONCLUSION

In this paper, we proposed a fast template matching algorithm that performs the winner-update search in the Walsh-Hadamard domain. The proposed algorithm exploits the energy packing capabilities of the Walsh-Hadamard transform to further increase the efficiency of the winner-update search. Compared to the previous projection-kernel based pattern search method, our algorithm does not require any threshold selection. Experimental results showed the computational speed of the proposed template matching algorithm is comparable to that of the projection kernel based method with the optimal threshold.

ACKNOWLEDGEMENTS

This work was supported in part by MOEA project under grant 95-EC-17-A-01-S1-034 and National Science Council, under grant 95-2220-E-007-028.

REFERENCES

[1] Y. Hel-Or and H. Hel-Or, “Real-Time Pattern Matching Using Projection Kernels,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, pp. 1430-1445, Sept. 2005.

[2] Y.S. Chen, Y.P. Huang, and C.S. Fuh, “A Fast Block Matching Algorithm Based on the Winner-Update Strategy,” *IEEE Trans. Image Processing*, vol. 10, 2000.

[3] M. Gharavi-Alkhansari, “A fast globally optimal algorithm for template matching using low-resolution pruning”, *IEEE Trans. on Image Processing*, Vol. 10, No. 4, pp. 526-533, April 2001.

[4] S. Zhu, and K.K. Ma, "A New Diamond Search Algorithm for Fast Block-Matching Motion Estimation," *IEEE Trans. Image Processing*, vol. 9, Feb. 2000.

[5] T. Koga, K. Iinuma, A. Hirano, Y. Iijima, and T. Ishinguro, "Motion Compensated Interframe Coding for Video Conferencing," Proc. Nat. Telecommun. Conf., Nov. 1981.

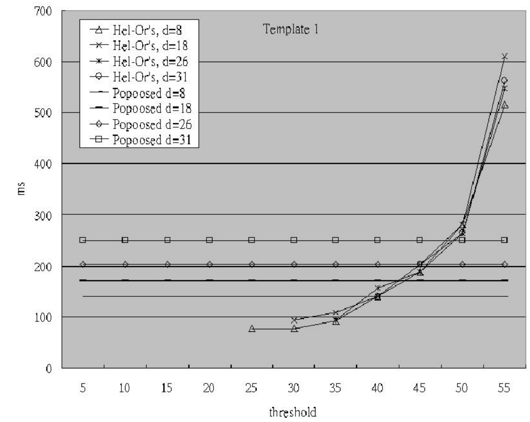
[6] R. Li, B. Zeng, and M.L. Liou, "A New Three-Step Search Algorithm for Block Motion Estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol 4, Aug. 1994.

[7] L.M. Po and W.C. Ma, "A Novel Four-Step Search Algorithm for Fast Block Motion Estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, June, 1996.

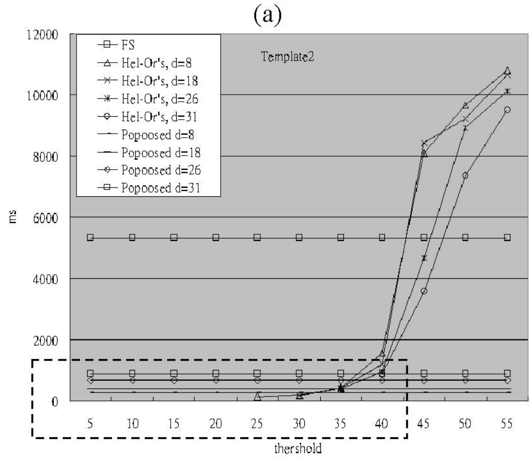
(a)

(b)

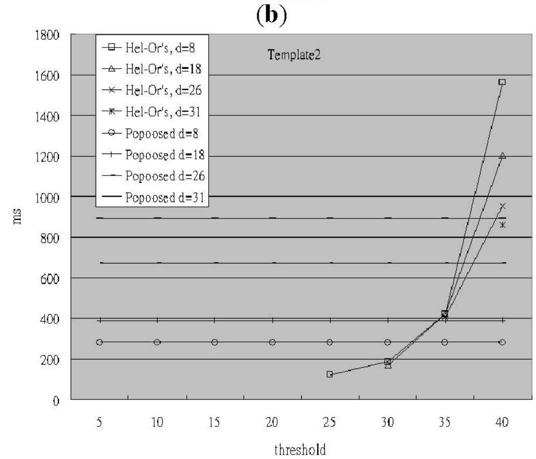
Fig 3: (a) (b) are the Lenna image added with Gaussian noises of standard deviation 18 and 31, respectively.



(a)



(**b**)



(c)

Fig. 4: The results of applying the proposed method and the method by Hel-Or and Hel-Or [1] on the experiments for Template(a) 1 and (b) 2 added with random noises of different standard deviations *d*’s. The enlarged version of the dotted rectangle of (b) is shown in (c).