Object Detection based on Fast Template Matching through Adaptive Partition Search

Wisarut Chantara and Yo-Sung Ho
School of Information and Communications
Gwangju Institute of Science and Technology
123 Cheomdangwagi-ro, Buk-gu, Gwangju 500-712, Republic of Korea
{wisarut, hoyo}@gist.ac.kr

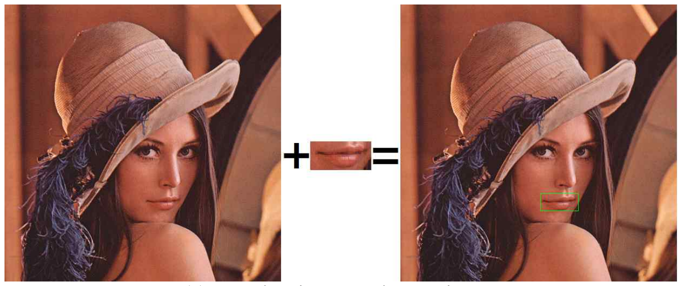
***Abstract*— In computer vision, object detection is one of the most researched topics. The goal of object detection is to detect all instances of objects from a known class, such as people, cars or faces in an image. Object detection uses the extracted features and learning algorithms to detect and recognize objects. In this paper, we propose a robust object detection method based on fast template matching. We apply an adaptive partition search to divide the target image properly. During this process, we can make efficiently match each template into the sub-images based on distortion measures. Finally, the template image is updated appropriately by an adaptive template algorithm. Experimental results show that the proposed method is very efficient and fast for object detection.**

Keywords*— *object detection; fast template matching; adaptive partition search; adaptive template algorithm;

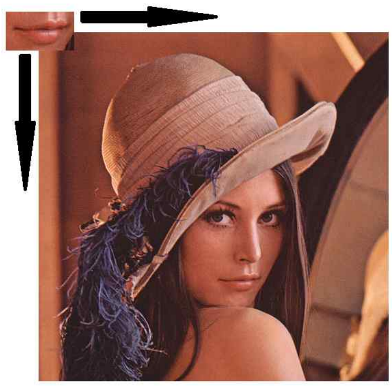
I. Introduction

Template matching is a technique used to find sub-images in the target image that matches the template pattern. This technique is widely used in object detection. The basic template matching algorithm calculates a distortion function that measures the degree of similarity between the template and the sub-image at each position of the image. Then, the position of the minimum distortion or maximum correlation is taken to locate the template into the examined image. The typical distortion measures used in template matching algorithms are the sum of absolute differences (SAD) [1] and the sum of squared differences (SSD) [2], while the normalized cross correlation (NCC) [3] and the zero-mean normalized cross correlation (ZNCC) [4-5] are by far the most widely used correlation measures. However, as far as template matching is concerned, NCC and ZNCC are often adopted for similarity measure as well. The traditional NCC and ZNCC need to compute the numerator and denominator which are very time-consuming. On the contrary, the conventional SAD and SSD are relatively simple.

In Fig. 1, in order to identify the matching area, the method needs to compare the template image against the target image by sliding it. By sliding the template, the process can measure the similarity between the template image and the region in the target image. At each location, a metric is calculated to represent the patch similarity between the images. The process then finds the location that yields the maximum correlation or the minimum distortion in the resulting image depending on the measurement.



(a) Template image and target image



(b) Window sliding and matching

Fig. 1. Template matching concept

A. *Template matching methods*

The conventional methods have been commonly used as metrics to evaluate the degree of similarity (or dissimilarity) between two compared images. The methods are simple algorithms for measuring the similarity between the template image (T) and the portions of the target image (I). Then, the process will classify the corresponding object.

The matching procedure calculates a resulting pixel using various matching methods. Equation (1) to Equation (3) show the matching techniques.

1) Sum of squared differences (SSD)

$$R(x,y)=\sum_{u,v}\left(T(u,v)-I(x+u,y+v)\right)^2 \tag{1}$$

SSD works by taking the squared difference between each pixel in *T* and the corresponding pixel in the portions of images being used for comparison in *I*. Squared differences are summed to create a simple metric of similarity.

2) Normalized cross correlation (NCC)

$$R(x,y)=\frac{\sum_{u,v}\left(T(u,v)\cdot I(x+u,y+v)\right)}{\sqrt{\sum_{u,v}T(u,v)^2\cdot\sum_{u,v}I(x+u,y+v)^2}} \tag{2}$$

NCC works by taking the product of each pixel in *T* and the corresponding pixel in the portions of images being used for comparison in *I*. The normalization process allows for handling linear brightness variation. The main advantage of NCC over the cross correlation is that it is less sensitive to linear changes in the amplitude of illumination in the two compared images.

3) Zero-mean normalized cross correlation (ZNCC)

$$R(x,y)=\frac{\sum_{u,v}(T^{'}(u,v)\cdot I^{'}(x+u,y+v))}{\sqrt{\sum_{u,v}T^{'}(u,v)^2\cdot\sum_{u,v}I^{'}(x+u,y+v)^2}} \tag{3}$$

where

$$T^{'}(u,v)=T(u,v)-\overline{T},$$

$$I^{'}(x+u,y+v)=I(x+u,y+v)-\overline{I}$$

ZNCC is even a more robust solution than NCC since it can also handle uniform brightness variation.

In addition, SSD technique needs to find the minimum value in the resulting image. The result of SSD function at a particular location is smaller, the local found sub-image is more similar to the searching template. If the result of SSD function is zero, the local sub-image is identical to the template. Otherwise, if using NCC or ZNCC, the process needs to find the maximum value in the resulting image.

Many studies on fast template matching methods have been reported, which focus on improving the search algorithm [6-7], reducing the number of matching points or based on two stage coarse fine approaches [8-10]. In regards to template matching, Banharnsakun and Tanathong [11] have introduced the use of the best-so-far ABC-based approach for object detection based on template matching by using the difference between the RGB level histogram corresponding to the target object and the template object. Even though Chantara's method [12] efficiently generates the template matching accuracy, the computational time of this method is high.

This paper proposes fast and efficient object tracking. In the first stage, a target image is accordingly remodeled following the interested object position, then standard robust matching techniques such as SSD, NCC, and ZNCC are applied to this image. This stage reduces the computational cost to a large extent. In the second stage, the object location is identified and the correct positions are updated properly in the whole target image, while the template image is adapted properly. This stage shows the efficiency of tracking method. Experiments show that the proposed method outperforms the traditional search algorithms.

II. Proposed Method

The proposed method is initially motivated by Chantara's work [12] based on matching accuracy. For practical use, we adapt this method. Based on Chantara's work, the proposed method modifies the object location and the adaptive template matching to improve the matching accuracy. Furthermore, the proposed method applied an adaptive partition zone search to enhance the computational time. Therefore, the contribution of the proposed method is to reduce computational cost and increase the matching accuracy. As the similarity measure, SSD, NCC, and ZNCC are the most popular and widely used for several applications. NCC and ZNCC also are more robust than SSD; nevertheless, NCC and ZNCC are more time-consuming than SSD. Therefore, the proposed method overcomes this issue.

The following two key processes are involved in the proposed method implementation. Firstly, the system detects and tracks the interested object in the object detection process. Finally, the template image is updated suitably on the object location and updating template process.

A. Object detection

In this subsection, we modify the conventional methods such as SSD, NCC, and ZNCC with an adaptive partition zone search. This algorithm decreases the computing time of the traditional algorithms.

a) Adaptive Partition Zone Search

An adaptive partition zone search reallocates the target image. This process is implemented as follows: (1) a previous area of the object (PA) in a target image is considered, (2) the target image is resized with suitable size following Fig. 2 while the center point of a new target image is PA, and (3) the new size of the target image (NI) is created. Therefore, NA will relate to a previous area of the object as follows Table I:

When the appropriate target image is provided as shown in Fig. 2, we perform object detection with template matching. The data result of matching algorithm provides the location of the corresponding object (ML) in the target image. Other candidate results and locations (CLs), which are in a limit of a threshold value, can also be found. These values are applied in the next subsection.

TABLE I. PARTITION ZONE SEARCH DETAILS

Assumption: The PA is stored in a memory buffer.

Zone	Descriptions	
Zone 0	**Condition**	Consider PA position in the target image
	Process	*1. Center of NI is PA position* *2. Create Zone 0 following Fig2.* *3. Set NI size = Zone 0 Size*

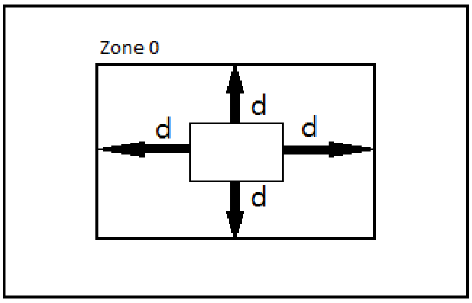


Fig. 2. Adaptive partition zone search

B. *Object location and updating template*

This subsection proposes a method for locating the appropriate area of interested object (IO) in a whole target image and updating the proper template image as follows Table II. This process is implemented as follows: (1) ML or CLs positions are considered with PA, (2) IO is assigned the proper location, (3) the previous template is updated with the current template, and (4) the current template is restored with a new template. There are three situations are applied to this process as follow Table II.

TABLE II. THE BEST OBJECT LOCATION PROCESS

Assumption: The PA is stored in a memory buffer.

Situation	Method procedure	
Situation 1	**Condition**	ML is the closest of PA
	Process	*1. Set IO = ML* *2. Previous template = Current template* *3.Current template = Original template*
Situation 2	**Condition**	CLs is the closest of PA
	Process	*1. Set IO = the closets of CLs* *2. Previous template = Current template* *3.Current template = Object template*
Situation 3	**Condition**	CL is not situation 1 and 2
	Process	*1. Set IO = PA* *2. Previous template = Current template* *3.Current template = Object template*

III. EXPERIMENT RESULTS

In this section, we show the computational advantage of the proposed method with the template matching algorithms, e.g., SSD, NCC, and ZNCC for comparison. The experiments are performed to examine the computational time and the matching accuracy. The system executes on a PC with an Intel(R) Core(TM) i7-3930K CPU 3.20 GHz, 16.0 GB RAM and operating system of Windows 8.1.

We test with four images sequence sets. These reference test data are *Walking2*, *Woman*, *FaceOcc1, and Face.* These data sets are referred by Visual Tracker Benchmark [13] except for *Face*. *Face* is referred by our laboratory. A 29 × 109 sized template image is used to match in target image sequences (*Walking2*) which are the size of 384 × 288 pixels, as shown in Fig. 3. Furthermore, other template images and the related target images contain different sizes and different illuminations are illustrated.

Fig. 3. (Left) Source image (Right) Template image of *"Walking2"* sequence

Fig. 4. (Left) Source image (Right) Template image of *"Woman"* sequence

Fig. 4 shows the template image of 27 × 67 pixels is applied to slide into target image sequences (*Woman*) 352 × 228 pixels. While Fig. 5 displays the 252 × 228 pixels of target image sequences (*FaceOcc1*) which are processed with the template image of 29 × 109 pixels. Moreover, the template image of 117 × 230 pixels and the corresponding target image (*Face*) which is 852 × 480 pixels are shown in Fig. 6.

Fig. 5. (Left) Source image (Right) Template image of *"FaceOcc1"* sequence

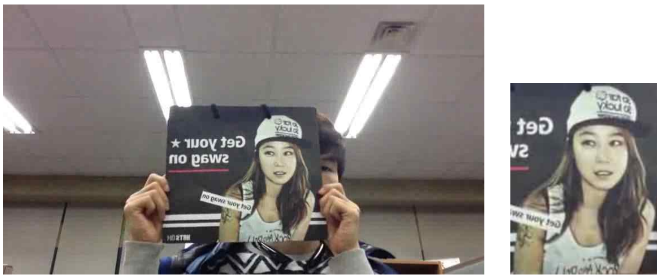


Fig. 6. (Left) Source image (852x480 pixels) (Right) Template image (177x230 pixels) of *"Face"* sequence

Fig. 7. Detected interested object result by the proposed method with SSD

The experiment outcomes with the proposed algorithm are implemented with SSD, NCC, and ZNCC. Fig. 7 illustrates the proposed method applied with SSD can detect the object properly with several image sequence data sets. Even though we apply the proposed method with other conventional methods such as NCC and ZNCC. The results show that the proposed method can specify the interested object position accurately as shown in Fig. 8 and 9 respectively. We have confirmed that the proposed method is efficient for template matching application.

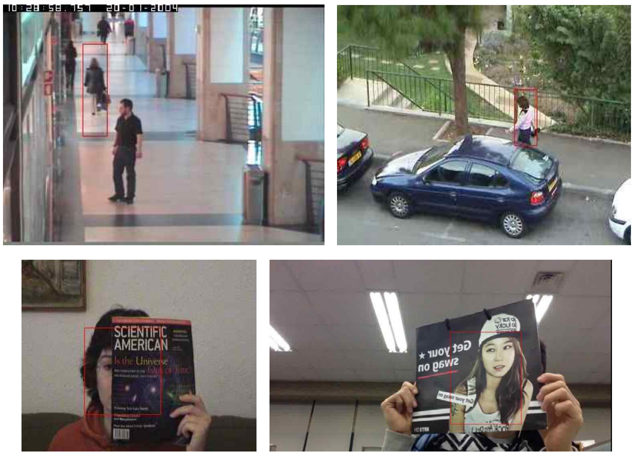


Fig. 8. Detected interested object result by the proposed method with NCC

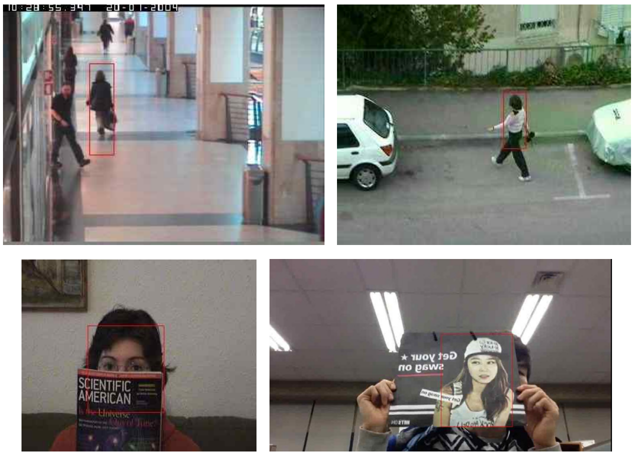


Fig. 9. Detected interested object result by the proposed method with ZNCC

Table III to Table V presents the objective evaluation which measures the computational time of algorithms SSD, NCC, and ZNCC are applied with the proposed method respectively. These results indicate that the proposed method outperforms other comparative methods by 58.59 ms, 58.28 ms, and 61.84 ms on average. These results show that the proposed method is highly effective in computational cost.

TABLE III. COMPUTATIONAL TIME [MS] USING ALGORITHMS BASED ON SSD METHOD AND PROPOSED METHOD

Image Sequence	**Computational time (ms)**		
	SSD	***Chantara's method [12]***	***Proposed method with SSD***
Walking2	78.84	78.44	33.63
Woman	76.62	76.20	31.22
FaceOcc1	63.75	63.35	45.77
Face	245.11	244.70	123.74
Average	**116.08**	**115.67**	**58.59**

TABLE IV. COMPUTATIONAL TIME [MS] USING NCC METHOD AND PROPOSED METHOD

Image Sequence	Computational time (ms)	
	NCC	***Proposed method with NCC***
Walking2	77.15	38.36
Woman	77.16	30.35
FaceOcc1	68.55	42.15
Face	245.63	122.26
Average	**117.12**	**58.28**

TABLE V. COMPUTATIONAL TIME [MS] USING ZNCC METHOD AND PROPOSED METHOD

Image Sequence	Computational time (ms)	
	ZNCC	***Proposed method with ZNCC***
Walking2	84.33	38.33
Woman	87.08	30.88
FaceOcc1	72.85	48.06
Face	264.02	130.08
Average	**127.07**	**61.84**

The ratio of the computational time (i.e. speed-up) can be calculated by

$$S = T\_old \,/\, T\_new \tag{4}$$

Where S is the ratio of computational time, T_old is the execution time of conventional method and T_new is the execution time of the proposed method.

TABLE VI. OBJECTIVE EVALUATION OF THE PROPOSED METHOD, COMPARING THE RATIOS OF THE COMPUTATINAL TIME WITH SSD, NCC, AND ZNCC METHOD

Image Sequence	Speed-up		
	Proposed method with SSD	***Proposed method with NCC***	***Proposed method with ZNCC***
Walking2	2.34	2.01	2.20
Woman	2.45	2.54	2.82
FaceOcc1	1.39	1.62	1.52
Face	1.98	2.01	2.03
Average	**2.04**	**2.05**	**2.14**

Table VI lists the ratios of computational time with respect to the conventional method (SSD, NCC, and ZNCC) The proposed method with SSD has provided a 2.04× speedup over original SSD method. While a 2.05× speedup of the proposed method with NCC is over the conventional NCC algorithm. Furthermore, The ZNCC proposed method has given a 2.14 × speedup over ZNCC method.

Table VII to Table X list the coordinate of the best matching location of these algorithms in *Walking2* image sequence, *Woman* image sequence, *FaceOcc*1 image sequence and *Face* image sequence, respectively. The results show that the proposed method is able to locate the correct matching position in all the image sequence. The main contribution of the proposed method, i.e., adaptive partition search zone, can be applied to other methods. The final quality depends on the performance of the algorithm that it is based on.

IV. CONCLUSIONS

In this paper, we proposed an object detection based on fast template matching using adaptive partition search. We apply an adaptive partition search to divide the target image. It gives the proper target image for the template matching algorithm. This process reduces the computational time. The appropriate target image has been taken to search the interested object by an adaptive template matching. This process provides an accurate object location on the target image. The proposed method also uses a small number of operations for the similar purpose when compared with conventional measures such as SSD, NCC, and ZNCC. So the proposed method decreases the computational cost and increases accuracy. Based on the experiment results, we can analyze that the proposed method has more efficient than the conventional methods like SSD, NCC, and ZNCC. Furthermore, a comparison of the interested object in the target image confirms that the proposed method outperforms the conventional methods.

TABLE VII. COMPARISON OF THE OBJECT'S POSITIONS WITH WALKING2 IMAGE SEQUENCE

Algorithm	Object's position (x, y)			
	Frame 1	***Frame 150***	***Frame 250***	***Frame 350***
SSD	130, 130	55, 36	122, 110	140, 90
Proposed method with SSD	130, 130	113, 83	101, 76	99, 51
NCC	130, 130	107, 75	122, 110	140, 90
Proposed method with NCC	130, 130	107, 76	95, 57	93, 46
ZNCC	130, 130	107, 75	122, 110	143, 90
Proposed method with ZNCC	130, 130	107, 76	98, 67	98, 52
Ground truth [13]	130, 132	111, 82	100, 65	93, 47

TABLE VIII. COMPARISON OF THE OBJECT'S POSITIONS WITH WOMAN IMAGE SEQUENCE

Algorithm	Object's position (x, y)			
	Frame 1	***Frame 50***	***Frame 100***	***Frame 200***
SSD	208, 114	43, 100	270, 90	256, 84
Proposed method with SSD	208, 115	212, 108	202, 115	222, 103
NCC	208, 114	212, 108	319, 83	45, 154
Proposed method with NCC	208, 114	212, 108	207, 113	213, 101
ZNCC	208, 114	212, 108	323, 88	47, 160
Proposed method with ZNCC	208, 115	212, 108	205, 112	213, 101
Ground truth [13]	213, 119	218, 110	206, 125	218, 104

TABLE IX. COMPARISON OF THE OBJECT'S POSITIONS WITH FACEOCC1 IMAGE SEQUENCE

Algorithm	Object's position (x, y)			
	Frame 1	***Frame 200***	***Frame 550***	***Frame 750***
SSD	118, 82	103, 101	96, 101	91, 83
Proposed method with SSD	119, 83	103, 102	96, 102	92, 84
NCC	119, 82	103, 101	5, 57	205, 0
Proposed method with NCC	119, 83	103, 102	95, 101	91, 85
ZNCC	119, 82	103, 101	6, 56	91, 84
Proposed method with ZNCC	119, 83	103, 101	96, 101	92, 84
Ground truth [13]	118, 69	113, 107	98, 109	88, 67

TABLE X. COMPARISON OF THE OBJECT'S POSITIONS WITH FACE IMAGE SEQUENCE

Algorithms	Object's position (x, y)			
	Frame 1	***Frame 120***	***Frame 240***	***Frame 400***
SSD	340, 173	50, 0	55, 0	50, 0
Proposed method with SSD	340, 174	404, 181	434, 167	432, 162
NCC	340, 173	404,181	51,0	51, 0
Proposed method with NCC	340, 174	404, 181	434, 167	436, 158
ZNCC	340, 173	404, 181	433, 161	437, 202
Proposed method with ZNCC	340, 174	404, 181	434, 166	431, 153

ACKNOWLEDGMENT

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Science, ICT & Future Planning(No. 2011-0030079).

REFERENCES

[1] J. P. Lewis, "Fast template matching," Vis. Inf., pp. 120-123, 1995.

[2] J. Shi and C.Tomisto, "Good feature to track," Proceedings of IEEE Computer Society Conference on Computer Vision Pattern Recognition, pp. 593-600, June 1994.

[3] W. K. Pratt, "Correlation techniques of image registration," IEEE Trans. On Aerospace and Electronic Systems, vol. AES-10, pp. 353-358, May 1974.

[4] L. D. Stefano, S. Mattoccia, and F. Tombari, "An algorithm for efficient and exhaustive template matching," In: Proc. Internation Conference on Image Analysis and Recognition, pp. 408-415, 2004.

[5] L. D. Stefano, S. Mattoccia, and F. Tombari, "ZNCC-based template matching using bounded partial correlation," Pattern Recognition Letters, vol. 26(14), pp. 2129-2134, Oct. 2005.

[6] H. Lio, C Xiao, M. Deng, and Zhifei Tang, "A faster image registration algorithm," 3rd International Congress on Image and Signal Processing (CISP2010),vol. 3, pp. 1218-1221, 2010.

[7] D. M. Tsai and C. T. Lin, "Fast normalized cross correlation for defect detection," Pattern Recognition Lett.,vol. 24, no. 15, pp. 2625-2631, 2003.

[8] G. J. Vanderburg and A. Rosenfeld, "Two-stage template matching," IEEE Trans. On Comput., vol. C-26, pp. 384-393, 1977.

[9] A. Goshtasby, S. H. Gage, and J, F. Bartholic, "A two-stage cross correlation approach to template matching," IEEE Trans. Pattern and Machine Intell., vol. PAMI-6, pp. 374-378, 1984.

[10] A. Rosenfeld and G. J. Vandenburg, "Coarse-fine template matching," IEEE Trans. Systtems, Man and Cybernetics, vol. 7(2), pp. 104-107, 1977.

[11] A. Banharnsakun and S. Tanathong, "Object Detection of Template Matching through Use of Best-So-Far ABC," Computational Intelligence and Neuroscience, vol. 2014(2014), pp. 1-8, 2014

[12] W. Chantara, J. H. Mun, D. W. Shin, and Y. S. Ho, "Object Tracking using Adaptive Template Matching," IEIE Transaction on Smart Processing and Computing, vol. 4(1), pp. 1-9, 2015.

[13] Y. Wu, J. Lim, and M.H. Yang, "Online Object Tracking: A Benchmark," Computer Vision and Pattern Recognition (CVPR), IEEE Conference on, pp. 2411-2418, June 2013.