A fast top-down visual attention method to accelerate template matching

Yiping Shen, Shuxiao Li, Chengfei Zhu*, Hongxing Chang

Institute of Automation Chinese Academy of Sciences, Beijing, China, 95 Zhongguancun East Road, Beijing, China

Received 1 March 2014, www.tsi.lv

Abstract

This paper presents a fast top-down visual attention method to downsize the search space of template matching. Such a method first generates patterns representing the local structures, and then calculates the pattern distributions representing the template and its surroundings. From here two separate operations are performed: the "pattern weight" is first introduced, which describes how well a certain pattern is correlated to the template, and then weights of all patterns are calculated for later reference. This is the "off-line" operation, and in comparison the "on-line" operation only calculates the pattern of each pixel, whose weights can be indexed conveniently from the off-line results. With all pixels' pattern weights calculated, the weight image is ready, from which we can extract the region of interest for subsequent matching. Experiments showed that our method obtained at least 6.21 times speed-ups over the state-of-the-art methods with little or no loss in performance.

Keywords: template matching, visual attention, top-down attention, saliency, region of interest

1 Introduction

Template matching (TM) is defined as searching for a sub-window (referred as candidate in the rest of this paper) that is most similar to a given template in a larger reference image. The similarity is usually measured as a cost function, e.g., product Cross Correlation (CC) [1], Normalized Cross Correlation (NCC) [1, 2], Zero-mean Normalized Cross Correlation (ZNCC) [1, 3], Sum of Absolute Differences (SAD) [4], Sum of Squared Difference (SSD) [5,6], Hamming Distance [7]. NCC and ZNCC are widely used due to their robustness to linear brightness variations [3].

The original algorithm of TM needs to search the entire space to get the best match, which is a time consuming procedure and thus limits its application in real-time environments not to mention on devices with limited computational resources. Much work has been explored to accelerate the computation of TM, which can be categorized into two aspects [2]. One aspect is to find an efficient representation of the template which enables fast computation of the cost function, e.g., Fast Fourier Transform (FFT) [1], Walsh-Hadamard transform [5], and Haar-like binary features [2]. Other techniques aim to reduce the search space or early prune the computation where the best match unlikely locates. For example, lower bound-based methods [5, 8] were used to accelerate the computation of SSD, while upper bound-based methods [3, 9-11] were used for NCC. Bound-based methods were more efficient, yet the order in which the candidates were examined affected the run time of the algorithms. A dual-bound method proposed in [6] obtained the best possible runtime by using a priority queue to determine an optimal ordering for examining the candidates. These efforts obtained high computation reduction. However, the run time of bound-based methods is data-dependent and may have no advantages over full space searching methods in the worst case.

Visual attention helps humans to fast focus on the information of interest when dealing with a huge mass of information [12, 13]. This property encourages researchers to bring it to machine vision systems. A considerable amount of research in cognitive science and computer vision has been conducted to understand and model visual attention mechanisms. Most of the research are concentrated on bottom-up attention (also called stimuli-driven) and relative models are built to analyse, which parts of the image attract human's attention in free viewing (for reviews please refer to [12, 14, 15]). However, the importance of the top-down (or task-driven) modulation have been emphasized in recent years [12, 16-18], and the integration of bottom-up saliency and top-down modulation models have been widely explored in [12, 16-19]. These models first computed the saliency maps based on the colour, intensity, and orientation features. Then top-down modulation was realized either by increasing the saliency on the expected location or increasing the weights of some specific features. Advances in visual attention are beneficial for solving some challenging problems in computer vision, e.g. object detection [16, 18, 19], tracking [20]. Nevertheless, some of these models involve time-consuming procedures, and current top-down models are mostly based on the bottom-up stimuli, which cannot deal with the situations in which the object

* ***Corresponding author*** e-mail: chengfei.zhu@ia.ac.cn

does not generate strong enough stimuli.

In this paper, a fast top-down visual attention method is proposed to reduce the search space of TM. The method consists of two parts: (1) a ROI is extracted based on the proposed top-down visual attention model; (2) ZNCC-based template matching is performed at each candidate of the ROI to get the final match. In the first part, the proposed method represents the local structure by patterns and builds pattern distributions for the template and the background, respectively. Note that, we can use a representative image containing the template (see Figure 1) or a set of images (see in Section 4.3) to empirically evaluate the pattern distribution for the background. If a representative image is used, we artificially warp the image and build pattern distributions with these warped images to obtain small scale invariance and in-plane rotation invariance. Then, pattern weights are calculated off-line by enhancing the template (referred as the specific object) patterns while suppressing the distracting background patterns simultaneously. These weights indicate how well the patterns are correlated to the specific object. For the on-line process, we only need to calculate the pattern for each pixel in the reference image and get the corresponding pattern weight by indexing in the learned model. This is the generation of the weight image. Then, the average weight of each candidate is computed by the integral image [21] and the one with highest average weight is the centre of the region of interest (ROI) which is extracted for subsequent template matching. The term "top-down" is referred because the generation of the weight image is controlled by the top-down knowledge, i.e. the appearance of the specific object showed in the template. Experiments show that our method obtains 30.90, 6.21, 24.08, 2.97 times speed-ups over a sequential implementation of FFTs [22], a state-of-the-art ZNCC-based method named Two-stage extended-mode Partial Computation Elimination (TPCE) [11], a state-of-the-art SAD-based method named Partial Distortion Elimination (PDE) [4] and the highly optimized implementation based on FFT in openCV (called HFFT for short, see in http://opencv.org), respectively with little or no lose in performance. The advantage and novelty of our method mainly include:

- Comparing to bound-based methods, the run time of the proposed method is data independent.
- We propose a top-down visual attention model to downsize the search space. In this model patterns represent local structures and the pattern weight describes how well a certain pattern is correlated to the template. Using patterns as the stimuli and pattern weights as the strength of stimuli, the top-down control is realized by setting pattern weights learned off-line.

This paper is organized as follows. Section 2 is an introduction of ZNCC-based TM. Section 3 describes the details of the proposed method. In section 4, we verify the method with experiments and compare it with four fast TM algorithms: FFTs [22], TPCE [11], PDE [4], and HFFT. Section 5 gives the conclusion of this paper.

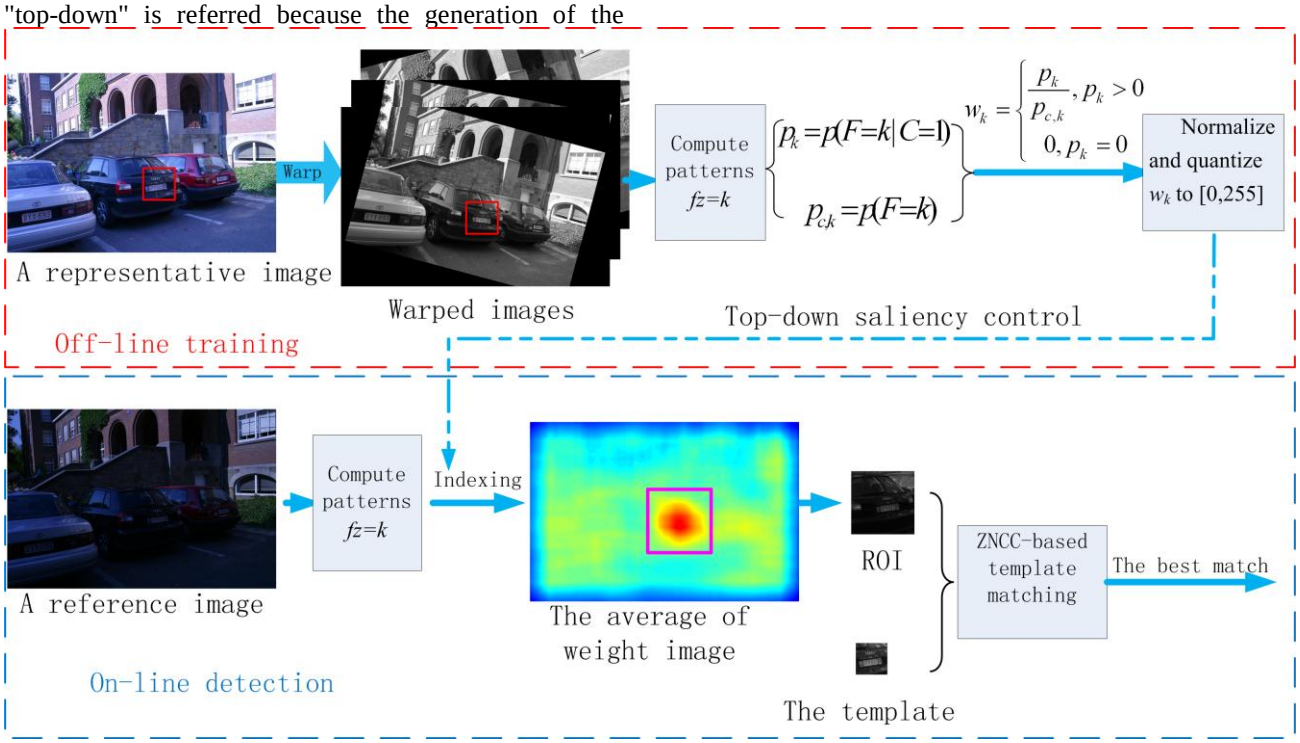

FIGURE 1 The flowchart of the proposed method. The red rectangle denotes the specify object to be detected (i.e. the template)

2 Template Matching Using ZNCC

Let I and T denote the reference image and the template, respectively. The size of I is $M \times N$ pixels, while the size of T is $m \times n$ pixels, where $m \le M$ and $n \le N$. The similarity between T and I at location (x,y) can be given by:

$$ZNCC(x,y)=\frac{\sum_{j=1}^{m}\sum_{i=1}^{n}[I(x+i,y+j)-\mu(x,y)]\cdot[T(i,j)-\mu(T)]}{\sqrt{\sum_{j=1}^{m}\sum_{i=1}^{n}[I(x+i,y+j)-\mu(x,y)]^2}\cdot\sqrt{\sum_{j=1}^{M}\sum_{i=1}^{N}[T(i,j)-\mu(T)]^2}}, \quad (1)$$

where

$$\mu(x,y)=\frac{1}{mn}\sum_{j=1}^{m}\sum_{i=1}^{n}I(x+i,y+j),\mu(T)=\frac{1}{mn}\sum_{j=1}^{m}\sum_{i=1}^{n}T(i,j). \quad (2)$$

The original full searching algorithm needs to scan the whole reference image and computes the ZNCC values for all candidates. Therefore, the total computation includes *mnJ* additions and *mnJ* multiplications, where $J=(M-m+1)\times(N-n+1)$ is the number of candidates. It can be reduced to $6MN\log_2(MN)$ additions and $6MN\log_2(MN)$ multiplications by using FFT [1].

3 The Proposed Top-Down Visual Attention Method

In this section, we first introduce the calculation of local structural patterns. Then, the proposed visual attention model is established by estimating and analysing pattern distributions for the template and the background using a representative image. At last, we describe the detection procedure from reference images based on the acquired visual attention model.

3.1 LOCAL STRUCTURAL PATTERN REPRESENTATION

Intensity, colour and orientation have been commonly used in visual attention computational models [12]. In this study, we use binary strings as an efficient representation of patterns and use patterns in our attention model. This idea comes from the Local Binary Patterns (LBs) [23], which own two advantages. First and most importantly, the feature space of LBs is a finite set, which enables us to establish a table to save the properties of patterns. Thus, once the properties (i.e. pattern weights) have been learned from the representative image off-line, we can get the weight of a certain pattern in the reference image by indexing. Secondly, LBs are more robust to illumination changes [23] than intensity and colour features, and more efficient than orientation features which often involve convolutions with Gabor filters.

LBs, first introduced by Ojala et al. [23], encode the pixel-wise information in an image, and have been widely used in texture classification [24] and face recognition because of its simplicity, efficiency, grayscale invariance and satisfactory discrimination [25]. LBs describe the relationship between the centre z_c and its P neighbours $z_0, z_1,\ldots, z_{P-1}$ (see Figure 2(a)). Formally,

$$LBP_{P,r}=\sum_{i=0}^{P-1}s(I(z_i)-I(z_c))2^i, \qquad s(x)=\begin{cases}1,x\ge 0\\0,x<0\end{cases}, \quad (3)$$

where $I(z_i)$ denotes the grayscale value at pixel z_i. A threshold t is used as follows [25] (t =3) to increase the robustness in flat areas:

$$LBP_{P,r}=\sum_{i=0}^{P-1}s(I(z_i)-I(z_c)+t)2^i. \quad (4)$$

In this study, we make some changes to the sampling points as follows: 16 points are sampled around the centre similar to the DAISY configuration [26] as shown in Figure 2(b). Two rings are used to make the local structural pattern more distinctive. Six points are sampled equally on the inner ring while ten points are sampled equally on the outer ring. The radius of the inner ring is r, while that of outer ring is $2r$. Experiments showed that setting r=5 can obtain the best performance. We use a Gaussian weighted sum of the grayscale value in the neighborhood instead of the grayscale value at pixel z_c and z_i, i = 0, 1, …,15 to deal with local distortions. The weighted sum is realized by a convolution with a 3×3 Gaussian kernel $[1\ 2\ 1]^T[1\ 2\ 1]$, where $[]^T$ is matrix transposition.

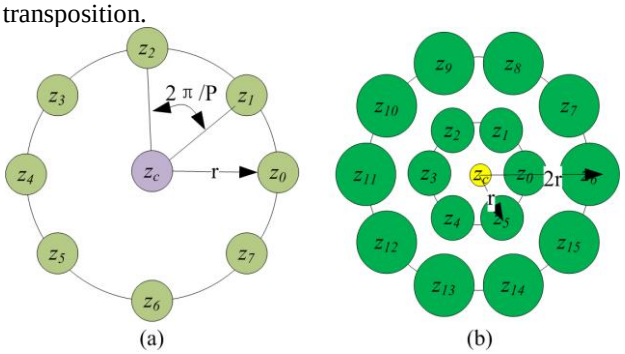

FIGURE 2 (a) The centre pixel z_c and its eight neighbours on the ring of r for P=8; (b) The configuration of DAISY_LBP. The small yellow circle denotes the centre; the large green circles denote 16 sampling points and the size of these circles corresponds to the smoothing range to deal with local distortions

The convolution only needs four additions and two multiplications for each pixel. Three convolutions with this kernel are used for sampling points on the outer ring, while two and one for points on the inner ring and the center point, respectively. Using (4), we get a 16-bit binary pattern f ($f\in[0,65535]$) termed as DAISY_LBP.

3.2 THE PROPOSED TOP-DOWN VISUAL ATTENTION MODEL

The basic insight of our model is that a pattern f gets

more saliency thus rewarding a higher weight if it takes place more frequently in the target than in background. Inspired by this insight and the saliency using natural image statistics model (SUN) [27] which performed well in predicting people's fixations in free viewing, we set up our top-down visual attention model. Let C=1 denote a point belonging to the target, C=0 denote that of the background, L denote the location of a pixel, and F denote the pattern of a pixel. Assuming that patterns and locations are independent, and conditional independent for given C=1, the saliency s_z can be defined as:

$$s_z = p(C=1|F=f_z, L=l_z) = \frac{p(F=f_z|C=1)}{p(F=f_z)} p(C=1|L=l_z), \tag{5}$$

where f_z is the pattern at pixel z. Since we have no priors about the location of the target, $p(C = 1|L = z)$ can be ignored in (5). So (5) can be rewritten as:

$$s_z = p(F=f_z|C=1) / p(F=f_z), \tag{6}$$

Using (6), we need to evaluate the pattern distributions for both the target and the background. The resulting saliency thus enhances the patterns of the target while it suppresses the patterns of distracting background. Note that useless target patterns are also suppressed if the background activates the same patterns more frequently.

In our work, the pattern distribution for the target is evaluated using the template, while the distribution for the background is estimated using a representative image or a set of images. Let p_k, k=0,1,… ,65535 be the probability of f_z=k in the target (the numerator in (6)), $p_{c,k}$ be the probability in the background (the denominator in (6)}), then the weight w_k of the pattern k=f_z can be calculated as:

$$w_k = \begin{cases} p_k / p_{c,k}, & p_k > 0 \\ 0, & p_k = 0 \end{cases}. \tag{7}$$

In this way, w_k implies the top-down control to the generation of the weight image.

We compute the whole set of w_k, k=0,1, …,65535 for a given template, and save them in a table during the off-line training phase. We also artificially warp the representative image to obtain small scale (0.85,1.15) and in-plane rotation(-15º,15º) robustness. 7 scale bins and 7 in-plane rotation bins are used in steps of 0.05 and 5º, respectively, yielding 49 warped images. At last, w_k, k=0,1, …,65535 is normalized and quantized to [0,255], which can be saved in a byte. The training phase is summarized in Table 1.

TABLE 1 Algorithms for off-line training and on-line detection

The off-line training algorithm	
Input: a representative image *Irep* and a template.	**Output: the top-down visual attention model W={w_k, |k=0,1, …,65535}.**
(1) Warp *Irep* using scale and in-plane rotation transform to get 49 images: $I_{rep,1}$, $I_{rep,2}$, …,$I_{rep,49}$.	
(2) For i=1,2, …,49, compute DAISY_LBP at every pixel in $I_{rep,i}$. Compute the histograms of patterns in the template $hist_t$ and the representative image $hist_r$.	
(3) For k=0,1, …,65535, compute p_k and $p_{c,k}$ from $hist_t$ and $hist_r$., and calculate w_k according to Equ. (7).	
(4) Normalize and quantize w_k to [0,255].	
The on-line detection algorithm	
Input: the reference image I_{ref}, and model W={w_k, |k=0,1, …,65535}	**Output: best matching position and score.**
(1) Compute DAISY_LBP at every pixel in image I_{ref}.	
(2) Assign w_{fz} to pixel z to generate the weight image.	
(3) A sliding window is run across the weight image to get the location (x_{opt}, y_{opt}) with maximum average weight by the integral image.	
(4) Extract the ROI.	
(5) HFFT is performed within the ROI to yield the final match.	

3.3 FAST DETECTION FROM REFERENCE IMAGES BASED ON THE SALIENCY MODEL

For the on-line detection phase, we first get the DAISY_LBP f_z for each pixel z in the reference image. Then, a weight image is generated by assigning w_{fz} to pixel z. Let the size of template be m×n. A sliding window of m×n is used to get the average weight of each candidate, which can be accelerated by the integral image. The candidate with the maximum average weight is considered as the centre of ROI, denoted by ($x_{opt,}$ y_{opt}). The size of ROI is decided according to the experimental results of the Euclidean distance between the ground truth and ($x_{opt,}$ y_{opt}), which will be discussed in section 4.1. Finally, HFFT is performed within the ROI to yield the final match. The on-line detection phase is summarized in Table 1.

Therefore, the computation of the on-line phase includes three convolutions with a 3×3 kernel (12*MN* additions and 6*MN* multiplications), the computation of patterns (16*MN* comparisons), the integral image (4*MN* additions), the average weight (4*MN* additions and *MN* comparisons) and TM within ROI ($6WH\log_2(WH)$ additions and $6WH\log2(WH)$ multiplications with the size of ROI $W\times H$ pixels). The proposed method eliminates $J\text{-}J_{ROI}$ candidates with an overhead of 20*MN* additions, 17*MN* comparisons and 6*MN* multiplications. Here J and J_{ROI} denote the numbers of candidates in the reference image and ROI, respectively. In comparison, FFT needs $6MN\log_2(MN)$ additions and $6MN\log_2(MN)$ multiplications.

4 Experimental results

4.1 EXPERIMENTS ON IMAGES WITH GAUSSIAN NOISE

Dataset. Forty images with size 640×480 are randomly chosen from MIT database (http://people.csail.mit.edu/torralba/images/), which is mainly concerned with indoor and urban scenes (see Figure 4). Five different levels of Gaussian noise with peak signal-to-noise ratio (PSNR) values of 27, 24, 21, 18, and 15 are added to each image of the dataset, respectively. Two template sizes 50×50 and 100×100 are used, and for each size 10 not-too-smooth templates are randomly chosen from each image. Therefore, there are 4000 matches in total (a match is defined as the most similar candidate found in a reference image).

Results. We evaluate the performance of the proposed method with different Gaussian noises as well as different configuration of P sample points described in Section 3.1. For P =8, the original LBs are employed; for P =12, the configuration of two rings with six points equally sampled on each ring is performed; and for P =16, the configuration is depicted in Figure 2(b). We compute the Euclidean distance d between the ground truth and the centre of the ROI (x_{opt}, y_{opt}), and draw the curve of $\#(d<x)/Total$ to x for each noise level as depicted in Figure 3. Here, $\#(d<x)$ denotes the number of matches with $d<x$ and *Total* (*Total* =40×10) denotes total matches in the experiment with the same noise level and P. We can see that (x_{opt}, y_{opt}) is closer to the ground truth with larger P. We do not investigate larger P than 16 (e.g. 32) because it needs too much memory for the weight tables. Thus, the suggested value of P is 16 for our method. When P =16, more than 75% and 89.5% of the total matches can be found whose distance is less than 10 pixels on sizes 100 × 100 and 50 × 50, respectively, and more than 98.25% and 99% of the total matches whose distance is lesser than 50 pixels on sizes 100×100 and 50×50, respectively. The size of ROI is set to be (m+99) × (n+99) for the following experiments according to this experiment, where $m\times n$ is the size of the template. Therefore, the number of candidates in ROI is 100×100=10000. Note that a smaller size of ROI contains less candidates in ROI thus leads to less computation. However, it may miss the most promising candidate.

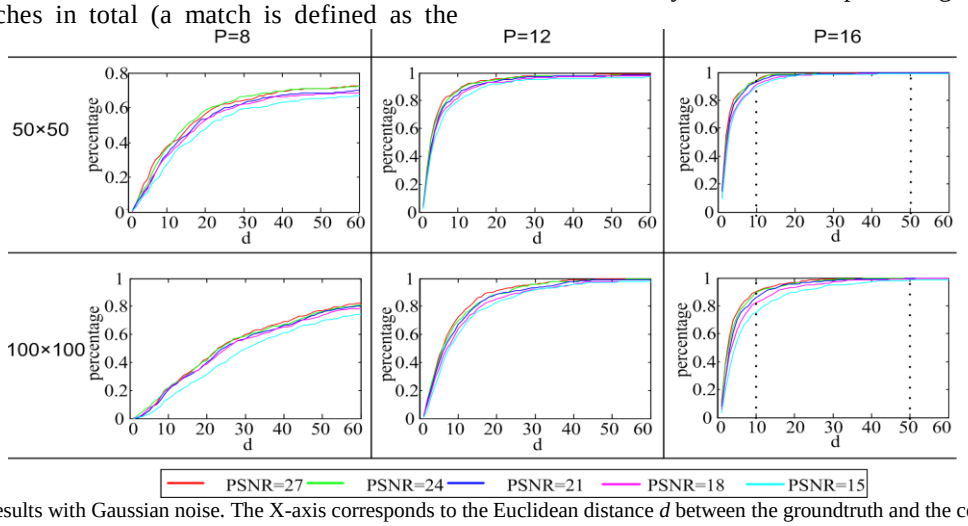

FIGURE 3 The results with Gaussian noise. The X-axis corresponds to the Euclidean distance d between the groundtruth and the center of extracted ROI (x_{opt}, y_{opt}). The Y-axis equals to $\#(d<x)/Total$, where $\#(d<x)$ denotes the number of matches with $d<x$ and *Total* (*Total* =40×10) denotes total matches in the experiment with the same noise level and P

4.2 EXPERIMENTS ON IMAGES WITH TRANSFORMATIONS

In this section, we compare our method with FFTs [22], TPCE [11], PDE [4] and HFFT. Demos for FFTs, TPCE and PDE are available at http://cvlab.lums.edu.pk/pce. The parameters of TPCE are set according to [11]. All algorithms are based on ZNCC except for PDE, which is a full search equivalent SAD-based algorithm. Five transformations were evaluated similar to [7]: small in-plane rotation, small scale changes, illumination changes, blur, and JPEG compression. All algorithms are implemented in C++ and run on an Intel Core2 Duo CPU E4400 2.00 GHz/2G RAM computer.

Dataset. The dataset is from OX database (http://www. robots.ox .ac.uk/~vgg/research/affine/). We use three groups of images (see Figure 4), which are

designed to test the robustness to illumination (Leuven), blur (Bikes), and Jpeg compression (Ubc). Six images in each group, we choose the first as the representative image and run the algorithms in the other five images. To evaluate the robustness to small geometrical changes, we create two data sets for scale and rotation changes. For small scale changes, Graffiti is warped to generate 10 images with scale randomly chosen in [-0.85, 1.15], and for in-plane rotation changes, Boat is rotated to yield 10 images with rotation angles in [-15º, 15º]. Therefore, we have five groups to evaluate these algorithms under the five transformations. For each group, five template sizes (32×32, 50×50, 64×64, 100×100, 128×128) are used and 40 templates are randomly chosen for each template size, yielding 7000 (5×(40×5×3+40×10×2)) matches. Templates with a standard deviation smaller than 60 are skipped to avoid the flat regions such as the blue sky in Ubc. Note that the templates are extracted from the representative image.

Results. Let "ROI_R", "$FINAL_R$" denote the results of ROI extraction and the final results of proposed method, respectively. We can easily obtain the location of the best match according to the true homography between the representative image and the reference image. The Euclidean distances d between the ground truth and the results by these algorithms are computed. We regard a match as a correct match if d is smaller than five pixels (meaning that the intersection of the detection and the ground truth exceeds 84% of the ground truth). The detection rate is defined as the number of correct matches with respect to the number of total matches. The speed-ups over FFTs in run time is defined similar to [11]. Results are showed in Figure 5. Note that the similarity threshold ρ_{th} of TPCE is empirically set to 0.9, meaning that TPCE will skip the candidates with a similarity smaller than 0.9, which explains the low detection rates of TPCE. A smaller ρ_{th} may increase both the detection rate and the run time. For example, setting ρ_{th} =0 will lead to full search equivalence. We do not consider smaller ρ_{th} but use the parameters in [11] if not specify. As showed in Figure 5, comparing with ZNCC-based algorithms, the detection rates of PDE are less than 0.38 for Leuven, indicating that SAD is not robust to illumination changes. Our method yields the same or very close detection rates as the full search ZNCC-based algorithms (FFTs and HFFT) for Leuven, Ubc and Graffiti, and performs better for Boat on sizes smaller than 100×100 indicating its robust to small in-plane rotation. However, our method obtains a lower detection rate than FFTs for Bikes on sizes larger than 50×50 because images with deep blur lose many textures and DAISY_LBPs are not good at discriminating local texture-less structures. The detection rate of our ROI extraction model for Boat is highest on sizes larger than 32×32 because the model is designed to be robust to small in-plane rotation ([-15º,15º]) while TM is not. For speed-ups over FFTs, our method obtains the highest speed-ups for our model can eliminate 97.78% of the candidates. HFFT obtains the second highest speed-ups except for Ubc on the size 32×32. The average computation elimination for TPCE and PDE are 90.18%, 73.40%, respectively, which explains that TPCE is faster than PDE. The average speed-ups of our method over FFTs, TPCE, PDE and HFFT are 30.90, 6.21, 24.08, 2.97 times, respectively.

FIGURE 4 Images from MIT database (the first row) and images from OX database (the second row)

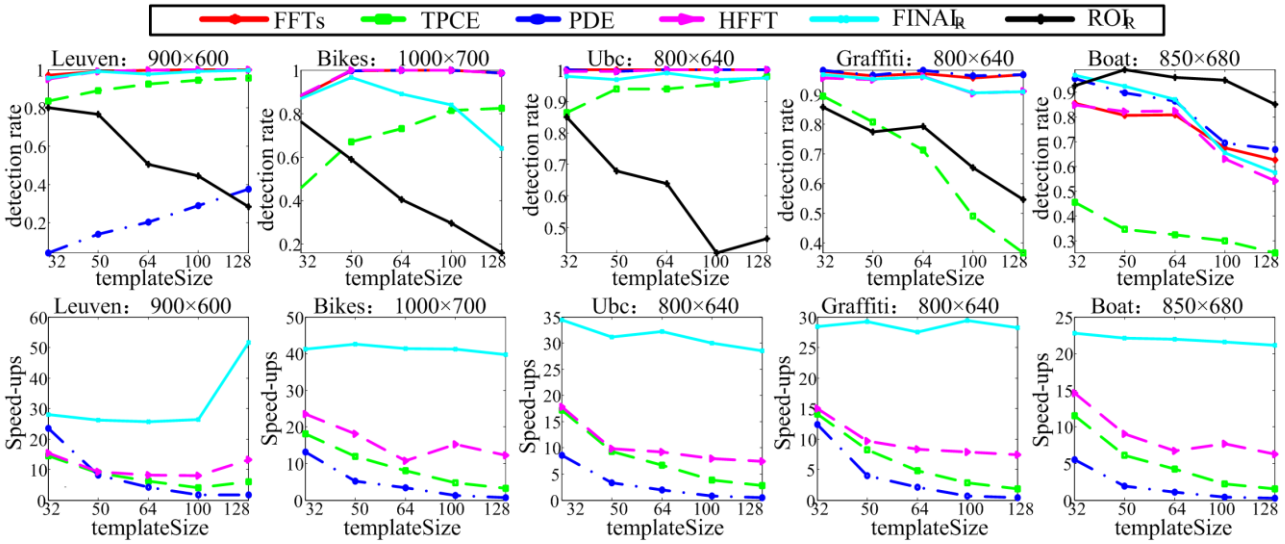

FIGURE 5 Detection rates and speed-ups over FFTs in run time for images with illumination changes (Leuven), blur (Bikes), JPEG compression (Ubc), small scale changes (Graffiti) and small in-plane rotation (Boat)

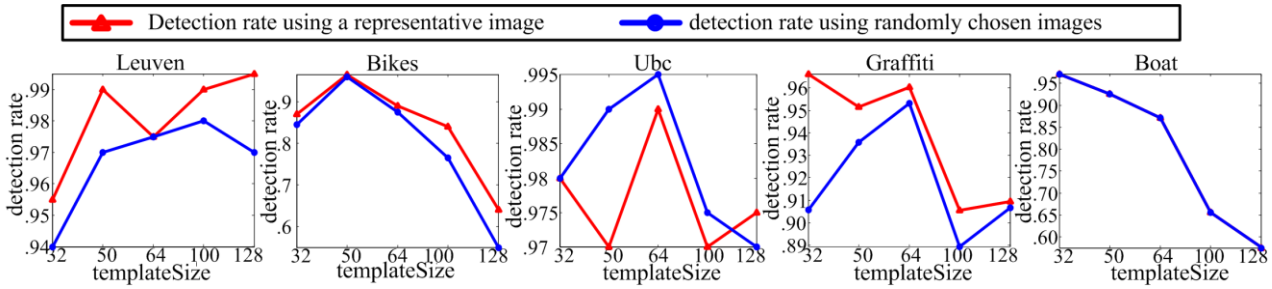

FIGURE 6 Comparison of the detection rate using a representative image and that using randomly chosen images to estimate the pattern distribution of the background

4.3 EXPERIMENTS WITH RANDOMLY CHOSEN IMAGES AS BACKGROUND

In the former experiments, a representative image is utilized to evaluate the pattern distribution of the background. However, there are cases that only a template is available. In this case, we can randomly choose a set of images to evaluate the pattern distribution of the background. In this experiment, 40 images mentioned in Section 4.1 are used to evaluate $p_{c,k}$, k=0,1,…,65535. We repeat the experiments in Section 4.2. The differences of detection rates are illustrated in Figure 6. Let "$pFINAL_{R1}$" and "$pFINAL_{R2}$" denote the detection rates in Section 4.2 and section 4.3 of the proposed method, respectively. As we can see, $pFINAL_{R2}$ are exactly the same as $pFINAL_{R1}$ for Boat, and very close to $pFINAL_{R1}$ for Leuven, Ubc, Graffiti and Bikes. Let $|x|$ denote the absolute value of x. The maximums of $|pFINAL_{R2}$-$pFINAL_{R1}|$ are 0.025, 0.090, 0.020, 0.060, 0.000 for Leuven, Bikes, Ubc, Graffiti and Boat, respectively. In all, using randomly chosen images to estimate the pattern distribution of the background does not have obvious influences on the performance of our method. Using the pre-computation of $p_{c,k}$, k=0,1,…,65535, the training phase only needs to compute p_k, which will further reduce the training time to less than 0.20 seconds for template size 128×128.

5 Conclusions

This paper proposes a fast top-down visual attention method to downsize the search space of TM. A texture pattern namely DAISY_LBP is first introduced, which is efficient to compute and robust to noise and local distortions. The pattern is used in the top-down visual attention model, and the pattern weight describes how well a certain pattern is correlated to the specific template. Using patterns as the stimuli and the pattern weights as the strength of stimuli, the top-down control is realized by setting the pattern weights learned off-line. Experiments show that our method obtains 30.90, 6.21, 24.08, 2.97 times speed-ups over FFTs, TPCE, PDE and HFFT, respectively with little or no loss in performance.

Our current method relies on a single ROI. In future work, several ROIs can be extracted to further improve the detection rate. The number of ROIs and the size of ROIs should make a compromise for efficiency. We will investigate the effects of these two terms. Efforts will also be given to the integration of colour and texture features into the algorithm for performance improvement.

Acknowledgments

This work is supported by National Natural Science Foundation of China (no. 61005028, no. 61175032 and No. 61101222).

References

[1] Lewis J P 1995 Fast Normalized Cross-Correlation *Vision Interface* **10**(1) 120-3
[2] Tang F, Tao H 2007 Fast Multi-Scale Template Matching Using Binary Features *IEEE Workshop on Applications of Computer Vision 2007* 36-41
[3] Stefano L D, Mattoccia S, Tombari F 2005 *Pattern Recognition Letters* **26**(14) 2129-34
[4] Montrucchio B, Quaglia D 2005 *IEEE Trans. on Circuits and Systems for Video Technology* **15**(2) 210-20
[5] Ouyang W, Tombari F, Mattoccia S., Di Stefano L, Cham W 2012 *IEEE Trans. on Pattern Analysis and Machine Intelligence* **34**(1) 127-43
[6] Schweitzer H, Deng R, Anderson R A 2011 *IEEE Trans. on Pattern Analysis and Machine Intelligence* **33**(3) 459-70
[7] Pele O, Werman M 2008 *IEEE Trans. on Pattern Analysis and Machine Intelligence* **30**(8) 1427-43
[8] Tombari F, Ouyang W, Stefano L D, Cham W 2011 *Pattern Recognition Letters* **32**(15) 2119-27
[9] Mattoccia S, Tombari F, Stefano L D 2008 *IEEE Trans. on Image Processing* **17**(4) 528-38
[10] Mattoccia S, Tombari F, Stefano L D 2011 *Pattern Recognition Letters* **32**(5) 694-700
[11] Mahmood A, Khan S 2012 *IEEE Trans. on Image Processing* **21**(4) 2099-108
[12] Borji A, Itti L 2012 *IEEE Trans. on Pattern Analysis and Machine Intelligence* **35**(1) 185-207
[13] Carrasco M 2011 *Vision Research* **51**(13) 1484-525
[14] Toet A 2011 *IEEE Trans. on Pattern Analysis and Machine Intelligence* **33**(11) 2131-46
[15] Duncan K, Sarkar S 2012 *IET Computer Vision* **6**(6) 514-23
[16] Frintrop S, Backer G, Rome E 2005 Goal-Directed Search with a Top-Down Modulated Computational Attention System *Proceeding of IEEE Conference on Computer Vision and Pattern Recognition* Berlin Heidelberg: Springer 117-24
[17] Navalpakkam V, Itti L 2006 An Integrated Model of Top-Down and Bottom-Up Attention for Optimizing Detection Speed *IEEE Conference on Computer Vision and Pattern Recognition 2006* **2** 2049-56
[18] Fang Y, Lin W, Lau C, Lee B 2011 A Visual Attention model Combining Top-Down and Bottom-Up Mechanisms for Salient Object Detection *IEEE International Conference on Acoustics, Speech and Signal Processing 2011* 1293-6
[19] Chang K, Liu T, Chen H, Lai S 2011 Fusing Generic Objectness and Visual Saliency for Salient Object Detection *IEEE International Conference on Computer Vision 2011* 914-21
[20] Ma L, Cheng J, Liu J, Wang J, Lu H 2010 Visual Attention Model Based Object Tracking *Advances in Multimedia Information Processing* Berlin Heidelberg: Springer 483-93
[21] Viola P, Jones M 2004 *International Journal of Computer Vision* **57**(2) 137-54
[22] William P, Saul T, William V, Brian F 1992 *Numerical Recipes. The Art of Scientific Computing* Cambridge:Cambridge University Press
[23] Ojala T, Pietikainen M, Maenpaa T 2002 *IEEE Trans. on Pattern Analysis and Machine Intelligence* **24**(7) 971-87
[24] Liu L, Zhao L, Long Y, Kuang G, Fieguth P 2012 *Image and Vision Computing* **30**(2) 86-99
[25] Pietikäinen M, Hadid A, Zhao G Y, Ahonen T 2011 *Computer Vision Using Local Binary Patterns*, London: Springer
[26] Winder S, Hua G 2009 Brown M Picking the Best Daisy, *IEEE Conference on Computer Vision and Pattern Recognition 2009* 178-85
[27] Kanan C, Tong M H, Zhang L, Cottrell G W *Visual Cognition* **17**(6) 979-1003

Authors

Yiping Shen

Current position, grades: PhD student of University of Chinese Academy of Sciences
University studies: University of Science and Technology of China (2005-2009).

Shuxiao Li

Current position, grades: Associate Professor, Institute of Automation, Chinese Academy of Sciences
University studies: Xi'an Jiao Tong University (1999-2003), PH.D. on Pattern Recognition and Intelligent System (2008, Institute of Automation, Chinese Academy of Sciences).
Scientific interest: computer science, machine vision, object recognition

Chengfei Zhu

Current position, grades: Assistant Professor, Institute of Automation, Chinese Academy of Sciences
University studies: University of Science and Technology of China (2000-2004), PhD on Pattern Recognition and Intelligent System (2010, Institute of Automation, Chinese Academy of Sciences
Scientific interest: computer science, object recognition

Hongxing Chang

Current position, grades: Professor, Institute of Automation, Chinese Academy of Sciences, Dean of the Integral Information Research Center
University studies: Beijing University of Aeronautics and Astronautics (1982-1986), Master (1991)
Scientific interest: Integral information processing and intelligent system, computer vision