

Wavelet Band-pass Filters for Matching Multiple Templates in Real-time

Yue Wu¹

ywu03@ece.tufts.edu

Pradeep Natarajan²

pradeepn@bbn.com

Joseph P. Noonan¹

jnoonan@ece.tufts.edu

Rohit Prasad²

rprasad@bbn.com

Premkumar Natarajan²

pnataraj@bbn.com

¹ Department of Electrical & Computer Engineering

Tufts University

161 College Ave.

Medford, MA 02155, USA

² Speech, Language & Multimedia Business Unit

Raytheon BBN Technologies,

10 Moulton St.

Cambridge, MA 02138, USA

Abstract

Most existing methods for template matching require computationally demanding searches and may not be appropriate for matching multiple templates. We present a novel real-time solution by matching a number of subtemplates with an input image, instead of using the entire template. Each template class is first learned and depicted offline as a set of wavelet-based directional band-pass filters, each of which allows only the most salient wavelet frequencies (subtemplates) to pass. As a result, matching a template becomes equivalent to the problem of filtering the wavelet transformed image using a set of band-pass filters. Finally, the region with the highest response density after filtering is considered as the detection. The computational complexity of this matching method is 1/50 of optimal correlation based SSD [1] and 1/10 of orthogonal Haar transform based SSD [2]. Further, the use of subtemplates enables detection even in the presence of skew and rotation. We present experimental results that demonstrate the capacity of our system for detection with a total of 10 different logo classes in broadcast videos.

1 Introduction

In many computer vision applications we are interested in matching a template with a given image to find the region of interest (ROI) that most closely matches the template in terms of some similarity measurement. According to the way similarity measurements are performed, such *template matching* methods can roughly be classified into two groups: 1) *patch matching* schemes, such as the sum of absolute difference (SAD) [3], the sum of squared difference (SSD) [4], or cross correlation (XCORR) [5], where the similarity measurement relies directly on pixel information from the patch of interest; and 2) *feature matching* schemes, such as invariant features [6, 7] and bags of features [8, 9, 10, 11], where similarity measurement relies on features describing the template and the frame.

Patch matching methods are not robust, especially when noise, skew, or errors occur [25]. Further, they consume a large amount of time [20], because of expensive sliding window search for calculating the similarity score over all possible locations. Several techniques have been explored for accelerating such matching methods, including early rejections [27] and correlation techniques [3, 5]. However, the computation cost could still be unaffordable when the frame size is large. Typically other techniques, like frame difference, are used to reduce the search space in applications.

Feature matching methods process the template and describe it with features [10, 17, 25], which are ideally invariant to rotation, skew, noise etc. However in many cases, the use of a more complicated model for similarity measurement results in higher computational cost. Further, sliding window search is also a costly stage for such methods [14]. While there exist known algorithms for fast search of object instances in an image using branch-and-bound techniques [13], in our particular problem, methods of this type have two crucial limitations. First, they require a large number of training samples for each class to learn robust classifiers. Second, interest point detectors like SIFT [16] typically do not generate sufficient number of feature points, because of the small size of the provided logo, large homogenous regions and degradations.

Wavelets based approaches that we consider in this paper have been extensively used in object detection and recognition. In [9], wavelet coefficients based image histograms are collected in bins and are used for classifying logos. In [18], wavelet coefficients are directly used for training pedestrian detectors. In [26], wavelet coefficients are selected to form rotation-invariant features by using the angular-radial transform. However, matching logos within frames using [9, 18, 26] still would require expensive window searching and thus are not appropriate for real-time processing.

In our work, we propose a new matching method using the wavelet based band-pass filters (WBPFs). Instead of using direct distance measurement, which requires expensive window search, the similarity is measured in an indirect way involving two stages. In the *offline template processing* stage, a template is automatically described using a set of three directional WBPFs, where only salient wavelet frequency components are allowed to pass. In the *online frame processing* stage, a frame is transformed to the wavelet domain and its sub-bands are filtered with respect to the corresponding template WBPFs. Finally, the detection is made at the region of densest responses under spatial constraints [8, 15]. We show that the proposed template matching system has a very low computational cost, which is 50 times faster than the correlation based SSD [5] and 10 times faster than the orthogonal Haar transform (OHT) based SSD [20]. Further, the proposed method does not trade-off accuracy, since the use of subtemplate information makes it robust to skew and camera view change. Experimental results demonstrate our method for real-time logo detection in broadcast videos.

2 Framework Overview and Wavelet Processing

2.1 Framework Overview

Figure 1 presents an overview of the matching framework using WBPFs. In the *offline template processing* stage, each template class is learned and described as a set of WBPFs, which only allows the frequency band containing salient subtemplate frequency components to pass. In the *online frame processing* stage, the frame is also transformed to the wavelet

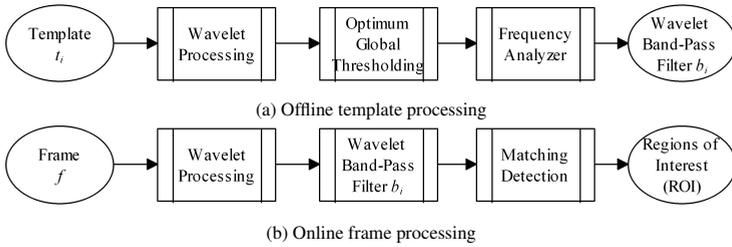


Figure 1: The proposed WBPFs multiple template matching framework

domain and it is filtered by WBPFs with respect to all template classes. For each template class, the filtered results are combined and analyzed to predict the region of interest (ROI). The offline template processing and the online frame processing are discussed in detail in future sections.

2.2 Why WBPFs works

We first present the general idea of template matching using the density response of subtemplates. Instead of treating all pixels in a template evenly, we discard non-salient ones and match based on the salient subtemplates. If we assume that the wavelet filter w is of length l , we transform the template to wavelet domain by convolving the wavelet filter coefficients with the template signal, as shown in Eqn. (1). More specifically, each coefficient in $W(t)$ is actually a convolution between a stripe of template pixels and the wavelet filter coefficients [8] by the definition of the wavelet transform.

$$\begin{cases} W(t) = t * h \\ W(t)[i] = \sum_{j=1}^l t[i-j] \cdot h[j] \end{cases} \quad (1)$$

Therefore, the wavelet transform expresses stripes of pixels (subtemplates) in the given template as coefficients. It is clear that not all subtemplates are salient and thus the selection stage should be performed. Using the alternative interpretation of wavelet coefficients in images [8], wavelet sub-bands can be considered as the high frequency response of a time domain signal. Therefore, the selection of salient templates is equivalent to choosing the most salient frequency responses in $W(t)$, which is well-known as power spectrum analysis in signal processing [8]. Consequently, a band-pass filter is then obtained, which covers the frequency band of all salient frequency components. Three wavelet sub-bands then imply three ways of obtaining subtemplates. In other words, the same template is described by three WBPFs. In online frame processing, the frame is transformed and then filtered by the template WBPFs. After filtering, only locations containing salient frequency components in the frame have responses. Further, a location is rejected unless it has responses in all three directions. Finally, the region with densest response is considered as the ROI.

2.3 Wavelet Transform and Related Issues

Noise is a crucial issue while processing real world data. To address this, the wavelet processing in both the offline template processing and the online frame processing in Figure 1, has the structure shown in Figure 2. It is worthwhile to note that in 'Wavelet Processing',

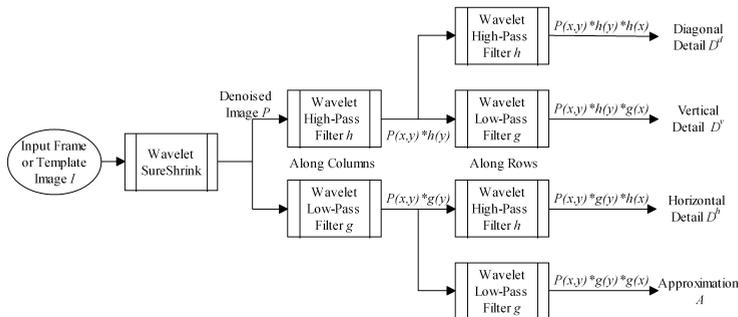


Figure 2: The internal structure of 'Wavelet Processing'

there are two major differences from a normal wavelet transform. First, the pre-processing stage of 'Wavelet SureShrink' [10] is applied to denoise the image. Second, no downsampling stage is used. The pre-processing is included since the image might be noisy (some of our data are converted from analog television broadcasts). The down-sampling stage is not included since we might lose significant matching information, especially when a spatial shift exists [13].

3 Offline Template Processing

3.1 Overview

The offline template processing is the core of the matching system, and provides the required band-pass filters for the online matching process. Figure 3 shows the detailed internal structures of this processing. The key challenge while using WBPFs for template matching is

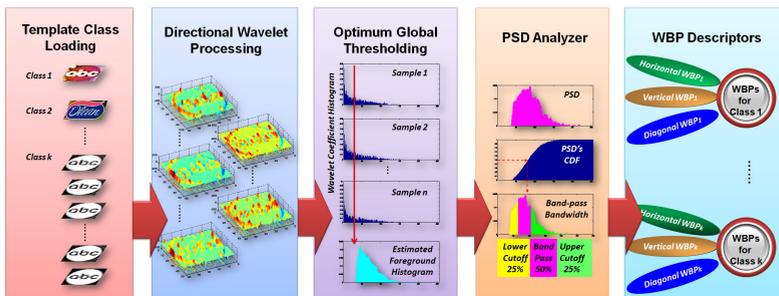


Figure 3: Offline template processing

the selection of cut-off frequencies. Ideally, a set of WBPFs should have the ability to accept template-like regions, and reject non-template-like regions. Thus, it is desirable to find the most representative wavelet coefficients for a template class. These coefficients should satisfy the following conditions:

- They should be distinctive from 'background' coefficients
- The number of representative coefficients should not be too large or too small.

The first condition eliminates 'background' coefficients, which appear widely in different template classes and thus are not representative. The second condition is empirical but

important, because the 'foreground' coefficients of extremely high values with small populations might be caused by random noise and thus are not representative; the 'foreground' coefficients of low values with large populations are from sidebands while convolving 'representative' peaks (See Figure 3-stage 2 'Directional Wavelet Processing'). The 'Optimum Global Thresholding' in Figure 3 helps eliminate 'background' coefficients and thus simplifies the problem. Further, the 'PSD Analyzer' finds 'foreground' coefficients with a moderate population and thus finds the corresponding cut-off frequencies.

3.2 Optimum Global Thresholding

Consider k samples from a template class given as t_1, t_2, \dots, t_k . Denote their corresponding wavelet coefficients along one direction as T_1, T_2, \dots, T_k . Let H_i be the i th wavelet coefficient histogram of $|T_i|$. In Figure 3-stage 3 'Optimum Global Thresholding', a given threshold r (marked as the red vertical line) separates the foreground and background coefficients on its right and left, respectively. Alternatively, for a given threshold r , the foreground and background coefficients in T_i are defined in Eqn. (2), where $N_i = \{1, 2, \dots, n_i\}$ and n_i is the number of coefficients in T_i .

$$\begin{cases} \text{Sample Foreground Class: } \mathcal{F}_i(r) &= \{j : |T_i[j]| \geq r \text{ and } j \in N_i\} \\ \text{Sample Background Class: } \mathcal{B}_i(r) &= \{j : |T_i[j]| \leq r \text{ and } j \in N_i\} \end{cases} \quad (2)$$

Then the foreground class and background class for this template are defined in Eqn. (3).

$$\begin{cases} \text{Template Foreground Class: } \mathcal{F}(r) &= \{\mathcal{F}_1(r), \mathcal{F}_2(r), \dots, \mathcal{F}_k(r)\} \\ \text{Template Background Class: } \mathcal{B}(r) &= \{\mathcal{B}_1(r), \mathcal{B}_2(r), \dots, \mathcal{B}_k(r)\} \end{cases} \quad (3)$$

As a result, the objective of Optimum Global Thresholding is to find an integer threshold r such that the inter-class variance is maximized. One solution is Otsu's method [19], which minimizes the intra-class variance. For k samples, the optimal threshold r^* is obtained by minimizing Eqn. (4):

$$r^* = \arg \min_{r \in [1, M]} \omega_{\mathcal{F}(r)} \sigma_{\mathcal{F}(r)}^2 + \omega_{\mathcal{B}(r)} \sigma_{\mathcal{B}(r)}^2 \quad (4)$$

where, $\omega_{\mathcal{F}(r)}/\omega_{\mathcal{B}(r)}$ and $\sigma_{\mathcal{F}(r)}/\sigma_{\mathcal{B}(r)}$ denote the numbers of coefficients and the standard deviation of class $\mathcal{F}(r)/\mathcal{B}(r)$, respectively; and M is the largest coefficient among k template samples, i.e. $M = \max\{|T_i[j]| : i \in \{1, 2, \dots, k\} \text{ and } j \in N_i\}$. Consequently, the estimated foreground histogram $H_{\mathcal{F}}$ (see 'Estimated Foreground Histogram' in Figure 3) is obtained by taking into account all sample foreground histograms as defined in Eqn. (5):

$$H_{\mathcal{F}}[j] = \begin{cases} 0, & \text{if } j < r^* \\ \sum_{i=1}^k H_i[j]/N_i, & \text{if } j \geq r^* \end{cases} \quad (5)$$

3.3 PSD Analyzer

'Optimum Global Thresholding' selects the 'foreground coefficients'. The last step is to select a number of representative coefficients. Heuristically, coefficients with large values are preferred. However, the number of the coefficients at this value also matters. Therefore,

it is natural to use the power spectrum density (PSD) analysis [6]. The raw PSD function p_{raw} is obtained by calculating the power contributions of coefficients. The PSD function p (see Figure 3-stage 4 'PSD Analyzer') is obtained by normalizing p_{raw} .

$$p_{raw}[j] = j^2 \cdot H_{\mathcal{F}}[j] \quad (6)$$

$$p[j] = p_{raw}[j] / \sum_{i=1}^M p_{raw}[i] \quad (7)$$

$$P[j] = \sum_{i=1}^j p[i] \quad (8)$$

Since $\sum_{j=1}^M p[j] = 1$ and $p[j] \geq 0$, the PSD function p is a probability density function. Correspondingly, its cumulative density function (CDF) P is defined in Eqn. (8). The cut-off frequencies are searched by using the following algorithm:

Algorithm 1 Searching cut-off frequencies for a template class

Input: the class PSD function p and its CDF function P

Output: the cut-off frequency pair c^L and c^U

Step 1. Find median location j^* , such that $j^* = \arg \min_{j \in \{1, 2, \dots, M\}} |P[j] - 0.5|$

Step 2. Set $c^L = c^U = j^*$

Step 3. Compare $P_{left} = P[c^L]$ with $P_{right} = 1 - P[c^U - 1]$

if $P_{left} > P_{right}$, let $c^L = c^L - 1$; otherwise, $c^U = c^U + 1$

Step 4. Repeat step 3, until $P[c^U] - P[c^L - 1] \geq 50\%$, i.e. at least 50% of power is covered

As a result, the cut-off frequencies c^L and c^U are obtained and stored for its corresponding template class as Figure 3 shows. Given a set of template samples belonging to the same class, repeat this process for corresponding wavelet coefficients along all three directions. Eventually, we obtain a set of three WBPFs for describing this template class.

4 Online Frame Processing

4.1 Overview

The flowchart of the online frame processing is given in Figure 4. For a given frame f , directional $W^H(f)$, $W^V(f)$ and $W^D(f)$ are obtained as the wavelet transform of f along the horizontal, vertical and diagonal directions, respectively. Later on, $W^H(f)$, $W^V(f)$ and $W^D(f)$ are filtered by the WBPFs corresponding to all template classes as stage 3 'WBPFs Filtering' in Figure 4 shows.

4.2 Matching Detection

Let $R_i^H(f)$, $R_i^V(f)$ and $R_i^D(f)$ denote the filtered results of frame f with respect to the i th template class along horizontal, vertical and diagonal directions respectively. Then the filtered result along direction X , $R_i^X(f)$, is defined as follows, where $X \in \{H, V, D\}$.

$$R_i^X(f)[j, k] = \begin{cases} 1, & \text{if } W^X(f)[j, k] \in WBPF_i^X \\ 0, & \text{otherwise} \end{cases} \quad (9)$$

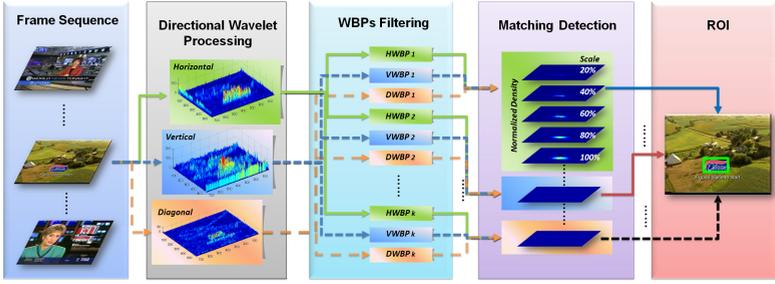


Figure 4: Online frame processing (Multiple template matching)

$$\mathcal{D}_i(f)[j, k] = \begin{cases} 1, & \text{if } R_i^H(f)[j, k] = R_i^V(f)[j, k] = R_i^D(f)[j, k] \\ 0, & \text{otherwise} \end{cases} \quad (10)$$

The sub-band filtering is equivalent to selecting locations that are similar to the sub-templates. These filtered results contain not only locations with actual logos but also false alarms. Fortunately, a great number of false alarms can be immediately eliminated by using the simple voting rule that each directional WBP has the veto to reject a coefficient location. In other words, a coefficient location will not be accepted for a template t unless all three WBPs agree as Eqn. (10) shows. Once $\mathcal{D}_i(f)$ is obtained, the density map at scale s is calculated as shown in Eqn. (11) by enumerating the number of responses in its neighborhood with respect to the corresponding rectangular template of width a_i and height b_i .

$$\rho_i^s(f)[j, k] = \sum_{y=-sb_i/2}^{sb_i/2} \sum_{x=-sa_i/2}^{sa_i/2} \mathcal{D}_i(f)[j+x, k+y] \quad (11)$$

According to the spatial randomness theorems [10], if the response density within a region of area Ω is λ , the appearance of v responses in a region of area A has the Poisson distribution given in Eqn. (12), when responses are uniformly distributed over Ω .

$$\Pr(\rho_i^s(f)[j, k] = v | \lambda) = \frac{(\lambda A)^v}{v!} \exp(-\lambda A) \quad (12)$$

$$\lambda = \sum_{j, k} \mathcal{D}_i(f)[j, k] / \Omega \quad (13)$$

In our case, A is the area of the template at scale s , i.e. $A = s^2 a_i b_i$, and Ω is the area of the frame. Then the normalized density score with respect to the spatially, uniformly distributed response is shown in Eqn. (14).

$$\overline{\rho}_i^s(f)[j, k] = 1 / \Pr(\rho_i^s(f)[j, k] | \lambda) \quad (14)$$

The detection scale and location (s^*, j^*, k^*) are chosen such that the maximum of $\overline{\rho}_i^s(f)[j, k]$ is achieved, i.e.

$$(s^*, j^*, k^*) = \arg \max_{s, j, k} \overline{\rho}_i^s(f)[j, k] \quad (15)$$

Finally, unless this maximum value is greater than a threshold Th , it is not considered as detection. Empirically, we found $Th = \Omega$, the area of the frame, to give good performance. This in effect enforces the condition that

$$\Pr(\rho_i^{S^*}(f)[j^*, k^*]|\lambda) \leq 1/\Omega \quad (16)$$

i.e. the observed response is much rarer than the probability of a random location in the frame having the correct match. Figure 5 shows the template and the corresponding intermediate results for stage 4 'Matching Detection' in Figure 4.

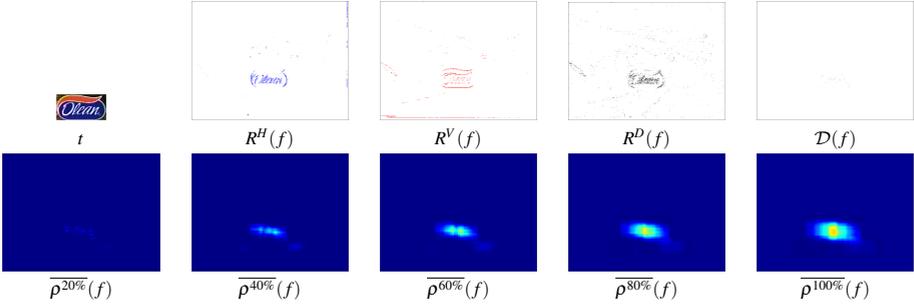


Figure 5: Intermediate results of 'Matching Detection'. First row: template and directional filtered results. Second row: normalized response density score for different template scales (brighter colors stand for higher scores).

4.3 Theoretical Algorithm Complexity

Assume the frame size is $M \times M$ and the used wavelet filters are of length l ($M \gg l$). Then matching k templates within one frame requires one wavelet transform with denoising for the frame image, $3k$ times band-pass filtering and additional $3k$ times voting. Table 1 shows the theoretical complexity of WBPFs for matching k templates.

Complexity\Operation	Wavelet Transform	Wavelet Denoising	WBPFs Filtering	Voting	Row Sum
Addition/Multiplication	$M^2[\mathbf{1}]$	$3M^2$	0	0	$4M^2$
Comparison	0	$3M^2$	$3kM^2$	$3kM^2$	$(3 + 6k)M^2$
Column Sum	M^2	$6M^2$	$3kM^2$	$3kM^2$	Total: $(7 + 6k)M^2$

Table 1: Theoretical WBPFs complexity (matching k templates in one frame)

Note that the complexity of $(7 + 6k)M^2$ can be largely reduced by using heuristic pruning techniques. For example, the band-pass filter can be accelerated using a hierarchical order, from the largest band-width to the smallest. Further, the band-pass filtered results are sparse and binary, and thus allow fast algorithms for matching detection.

Algorithm\Reported Complexity	One Template	k Templates (when k is large)
Optimal SSD Correlation [B]	$3C_{SR_{fft}}$	$\sim (22.656 \cdot \log_2 M)kM^2$
M4 SSD Correlation [B]	$5C_{SR_{fft}}$	$\sim (37.76 \cdot \log_2 M)kM^2$
OHT SSD [B]	$(4 + 7 \log_4 u)M^2$	$\sim (4 + 7 \log_4 u)kM^2$
WBPFs	$13M^2$	$\sim 6kM^2$

$$* C_{SR_{fft}} \approx 0.944 \cdot (4M^2 \log_2 M^2 - 6M^2 - 8) \quad [\square]$$

** M is the frame size, u is the number of Haar bases and l is the length of wavelet filter

Table 2: Theoretical arithmetic complexity comparison for fast matching algorithms

Table 2 shows the arithmetic complexity comparison table for recent fast matching algorithms. Table 3 extends the comparison table in [9]. When the frame size is of 512×512 and template size is of 64×64 , even for the one template and one frame case, WBPFs have only 2% of the computation for optimal SSD Correlation [9], which is the fastest of the algorithms listed in [9] and 10% of OHT SSD [20]. WBPFs produce even greater speed-up while matching multiple templates.

Method\Frame size	Template size 32×32			Template size 64×64		
	64×64	128×128	256×256	128×128	256×256	512×512
SSD Full Search	4,460,544	38,539,264	207,360,000	69,222,400	610,287,610	3,303,030,780
SSD Correlation[9]	500,574	2,373,798	10,980,849	2,373,798	10,980,849	49,865,529
SAD Full Search	3,345,408	28,904,448	155,520,000	51,916,800	457,715,710	2,477,273,080
SAD Correlation[9]	834,290	3,956,330	18,301,415	3,956,330	18,301,415	83,109,215
M4 Full Search	6,690,816	57,808,896	311,040,000	103,833,600	915,431,420	4,954,546,170
M4 Correlation[9]	834,292	3,956,334	18,301,416	3,956,330	18,301,415	83,109,215
OHT SSD[20]	159,744	638,976	2,555,904	753,664	3,014,656	1,2058,624
WBPFs	53248	212,992	479,232	212,992	479,232	1,331,200

Table 3: Comparison of number of computations required when matching one template with one frame at one scale

5 Experiments

We tested our approach for detecting 10 logo classes (one sample for each class) from broadcast news videos. For each logo class, 5 frames containing this logo were used as the test set. Figure 6 shows the template set and sample frames in the test set. We compared our approach with SSD [9] and M4 [9]. The method SSD Correlation [9] is selected for its simplicity of realization and the method of OHT SSD [20] produces equivalent matches to other SSD algorithms. Both the accuracy and speed are evaluated. Experiments were done on a Windows XP system and MATLAB r2010a environment with 3GB memory and Intel Core2 2.6GHz CPU.



Figure 6: Experiment logo data. Ten template classes are shown on the left half. Frames are directly pulled from broadcast videos (Note: logos might be skewed or scaled).

Figure 7 shows some results after matching. It is clear that the WBPFs system has the capacity to deal with skewed logos. This is because image data is of high information redundancy, whose neighbor pixels are closely correlated, and thus the wavelet coefficients also change within a small interval when a logo in the frame is slightly skewed. As a result, these comparable coefficients lead to match a skewed logo within a frame. We also noticed that the WBPFs system tends to return multiple detections when the logo size is small. This is because the wavelet coefficient histogram tends to have fewer coefficients and leads to a rougher estimator of the template class.



Figure 7: Template matching results using the proposed WBPFs system. The system is able to deal with skewed logos (see 'Windex' column), noisy logos (see 'Sprint' column), and scaled logos (see 'Sprint' column).

Table 4 shows the performance of SSD correlation, M4 and WBPFs on the user provided logo set shown in Figure 6. We consider a 'hit', if the marked ground truth region overlaps at least 50% of the detection region. It is clear that WBPFs outperforms fast algorithms SSD Correlation and M4 Correlation in both accuracy and speed.

Methods/Items	Precision	Accuracy		Matching Time (sec)		
		Recall	F-measure	1 template per frame	10 templates per frame	Total
SSD [8, 10]	0.4541	0.5600	0.5015	2.0974 ± 0.04071	20.9742 ± 0.40712	1055.11
M4 [9]	0.3901	0.5600	0.4598	3.8174 ± 0.07411	38.1744 ± 0.74114	1908.72
WBPFs	0.5322	0.9000	0.6689	0.3594 ± 0.00324	0.5763 ± 0.00722	28.82

Table 4: Performance of SSD Correlation, M4 and WBPFs on the user-provided logo set

6 Conclusions

We have proposed a new template matching framework where instead of matching an entire template within a frame, subtemplates are used. We generate a great number of subtemplates by using the wavelet transform along different directions. Then salient subtemplates are selected and further expressed as band-pass filters in the wavelet domain. As a result, matching a template within a frame becomes a problem of filtering the frame in the wavelet domain. Locations with responses after filtering the frame are those that match the subtemplates. Finally, the corresponding response density is calculated and the scale and location of the detection is determined by maximizing the normalized density score in Eqn. (14).

The proposed method has very low complexity and only requires one time wavelet transform, a finite amount of filtering, and user defined scale searches. Experimental results show that the proposed matching system is robust to skew, rotation, viewpoint change, scaling, and noise. In general, less than 500 locations survive after filtering a frame of size 528×704 with respect to all three WBPFs. Therefore, both response density calculations and scale searches can be done quickly. The proposed system can also work as a pre-processing step for efficiently reducing the search space and thus makes it possible to use expensive matching algorithms for real-time processing.

Acknowledgement: This paper is based upon work supported by the DARPA MADCAT Program. . The views expressed are those of the author and do not reflect the official policy or position of the Department of Defense or the US Government. We would like to thank Dr. Sangeetha Somayajula for her help with wavelet theory.

References

- [1] M. J. Atallah. Faster image template matching in the sum of the absolute value of differences measure. *IEEE Transactions on Image Processing*, 10(4):659–663, 2001.
- [2] David L. Donoho and Iain M. Johnstone. Adapting to unknown smoothness via wavelet shrinkage. *Journal of the American Statistical Association*, 90(432):1200–1224, 1995.
- [3] F. Essannouni and D. Aboutajdine. Fast frequency template matching using higher order statistics. *IEEE Transactions on Image Processing*, 19(3):826–830, 2010.
- [4] F. Essannouni, R. Thami, D. Aboutajdine, and A. Salam. Adjustable sad matching algorithm using frequency domain. *Journal of Real-Time Image Processing*, 1(4):257–265, 2007.
- [5] F. Essannouni, R. Oulad Haj Thami, D. Aboutajdine, and A. Salam. Simple noncircular correlation method for exhaustive sum square difference matching. *Optical Engineering*, 46(10):107004–4, 2007.
- [6] RC Gonzalez and RE Woods. *Digital image processing*. Pearson/Prentice Hall, 2008.
- [7] Choe Gwangwoo and Jr. Swartzlander, E. E. Merged arithmetic for computing wavelet transforms. In *Proceedings of the 8th Great Lakes Symposium on VLSI*, pages 196–201, 1998.
- [8] B. Hasan and D Hogg. Segmentation using deformable spatial priors with application to clothing. In *Proceedings of the British Machine Vision Conference*. BMVA Press, 2010.
- [9] A. Hesson and D. Androutsos. Logo classification using haar wavelet co-occurrence histograms. In *Proceedings of the Canadian Conference on Electrical and Computer Engineering*, pages 000927–000930, 2008.
- [10] S. Hinterstoisser, O. Kutter, N. Navab, P. Fua, and V. Lepetit. Real-time learning of accurate patch rectification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2945–2952. IEEE Computer Society, 2009.
- [11] J. Illian. *Statistical analysis and modelling of spatial point patterns*. John Wiley, 2008.
- [12] S. G. Johnson and M. Frigo. A modified split-radix fft with fewer arithmetic operations. *IEEE Transactions on Signal Processing*, 55(1):111–119, 2007.
- [13] C. H. Lampert, M. B. Blaschko, and T. Hofmann. Beyond sliding windows: Object localization by efficient subwindow search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

- [14] T. Lee and S. Soatto. Learning and matching multiscale template descriptors for real-time detection, localization and tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE Computer Society, 2011.
- [15] Hongsheng Li, Edward Kim, Xiaolei Huang, and Lei He. Object matching with a locally affine-invariant constraint. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1641–1648. IEEE Computer Society, 2010.
- [16] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.
- [17] P. Natarajan and R. Nevatia. View and scale invariant action recognition using multi-view shape-flow models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE Computer Society, 2008.
- [18] M. Oren, C. Papageorgiou, P. Sinha, E. Osuna, and T. Poggio. Pedestrian detection using wavelet templates. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 193–199. IEEE Computer Society, 1997.
- [19] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man and Cybernetics*, 9(1):62–66, 1979.
- [20] Wanli Ouyang, Renqi Zhang, and Wai-Kuen Cham. Fast pattern matching using orthogonal haar transform. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3050–3057, 2010.
- [21] Wang Quan and You Suya. Feature selection for real-time image matching systems. In *Proceedings of the International Conference on Pattern Recognition*, pages 1–4, 2008.
- [22] Marçal Rusinol and Josep Lladós. Logo spotting by a bag-of-words approach for document categorization. In *Proceedings of the International Conference on Document Analysis and Recognition*, pages 111–115. IEEE Computer Society, 2009.
- [23] I.W. Selesnick, R.G. Baraniuk, and N.C. Kingsbury. The dual-tree complex wavelet transform. *IEEE Signal Processing Magazine*, 22(6):123–151, 2005.
- [24] G. Takacs, V. Chandrasekhar, S. Tsai, D. Chen, R. Grzeszczuk, and B. Girod. Unified real-time tracking and recognition with rotation-invariant fast features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 934–941, 2010.
- [25] S. Taylor and T. Drummond. Multiple target localisation at over 100 fps. In *Proceedings of the British Machine Vision Conference*. BMVA Press, 2009.
- [26] Du-Ming Tsai and Cheng-Huei Chiang. Rotation-invariant pattern matching using wavelet decomposition. *Pattern Recognition Letters*, 23(1-3):191–201, 2002.
- [27] Shou-Der Wei and Shang-Hong Lai. Fast template matching based on normalized cross correlation with adaptive multilevel winner update. *IEEE Transactions on Image Processing*, 17(11):2227–2235, 2008.