

OBJECT DETECTION AND LOCALIZATION IN THE WAVELET DOMAIN

**Srivatsan Kandadai, Graduate Student
Klipsch School of Electrical and Computer Engineering
New Mexico State University
Las Cruces, NM 88003-88001**

ABSTRACT

We are interested in the problem of detecting and localizing objects in the compressed domain. The practical uses of this research are video surveillance, queries over digital library archives and teleconferencing. Most image operations, such as object recognition, are formulated as sequences of operations in the image domain. Such methods need direct access to pixel information as a starting point, but pixel information is not directly available in a compressed image stream. The standards that have emerged for still-image and video compression each contain steps that are commonly found in compression algorithms, like linear transformations, coefficient quantization, run-length coding and entropy coding. Coders like JPEG 2000 and SPHIT are built around the wavelet transform. Thus as a step toward detection and localization of objects embedded in the compressed bit stream we consider here the problem of localizing and detection in the wavelet domain.

INTRODUCTION

Content analysis of digitally transmitted video data typically has been performed on data either before it is compressed or after it is decompressed [1]. It may be more efficient, however to analyze and process the data as close to the fully compressed stage as possible. Compression algorithms create a set of domains in which the data resides, and the data moves through these domains according to well-defined transformations. The chain of compression steps defines the domains in which the data can be manipulated.

The advantage of operating on compressed data is the efficiency of eliminating some or all the processing related to decompression. Note that information present in a compressed stream is equivalent to what is obtained after decompression; the only difference is that they are represented differently. Consider a computer in a network whose task is to analyze an incoming compressed video stream and then pass the stream on to another computer. If the computer performs object classification in the pixel domain, it is forced to decompress the stream, perform the analysis, and then compress the stream again before sending it along on the network.

Another advantage to compressed domain processing is that certain problems can actually take advantage of the information that is made explicit by the compression algorithm. For example, compression schemes based on the frequency domain make the frequency information explicit. Coders like JPEG 2000 and SPHIT are built around the wavelet transform [2], [3]. Thus as a step toward detection and localization of objects embedded in the compressed bit stream we consider here the problem of localizing and detection in the wavelet domain.

THE WAVELET DOMAIN

Most compression schemes use similar approaches to compact data. They consist of the following key steps: (1) pixel range shifting; (2) decorrelation of data using transforms like the Karhunen Loeve Transform, the Discrete Cosine Transform or the Wavelet Transform; (3) quantization of the transform coefficients; (4) Run-length coding or Entropy coding. In this work we consider that the transform used for Decorrelation of data is the Wavelet Transform (WT) [4].

Wavelets are building blocks that can quickly decorrelate data. Decorrelation implies that; the representation of data in terms of wavelet coefficients is more “compact” than the original representation. In most images samples are close spatially are correlated, and signal energy is compacted into a small number of frequencies bands. To analyze and represent such signals, wavelets are needed which are local in space and frequency. Traditional wavelet transforms are developed using the filter bank method as shown in Fig. 1.

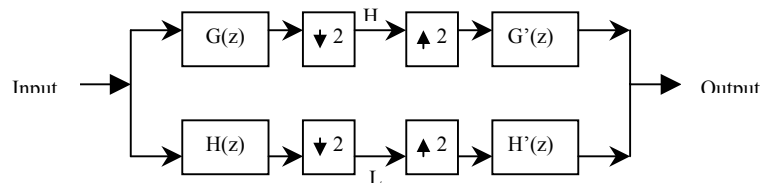


Figure 1 Filter bank Implementation of the wavelet transform. H denotes the high frequency components and L denotes the low frequency components.

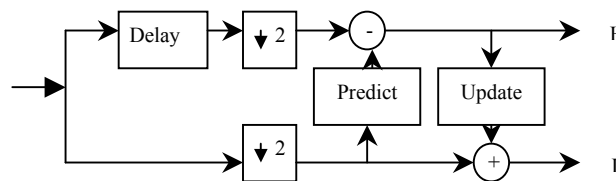


Figure 2. Lifted Implementation of the wavelet transform. H denotes the high frequency components and L denotes the low frequency components.

But in this work we use the lifted scheme [5] of implementing the wavelet transform. The advantages of implementing the wavelet transform using this scheme are: (1) the lifting scheme allows a in-place calculation of the wavelet transform; (2) with the lifting wavelet the inverse transform can be easily found by reversing the operations of the forward transform, in practice this comes down to simply reversing the order of the operations and changing each + to – and vice versa;

(3) the wavelet coefficients are located corresponding to the original position of pixels in the image. The lifting scheme is as shown in Fig. 2.

The lifted wavelet transform facilitates direct correlation of templates with the image, as the coefficient position in the transform domain is based on the position of the pixel in the original image. The arrangement of wavelet coefficients for one level of decomposition is as given in Fig.3. The high frequency components are denoted as H and the low frequency components are denoted as L, two symbols placed together like HL means, that low pass filtering was performed after high pass filtering.

From Fig.3 we see that there is a periodic shift variation by a factor of 2, and that the pixel energy is spread out into neighboring pixel values. Increase in depth of decomposition increases the shift variance by a factor of 2. Therefore for a depth of decomposition of 3 gives a shift variance of 8 pixels, and a depth of decomposition of 4 gives a shift variance of 16. This shift varying property of the lifted wavelet transform leads to errors in the detection of objects in the wavelet-transformed image. These errors can be countered by using multiple shifted templates. The idea behind multiple templates is to create insensitivity to different relative shifts.

Original Image:

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

Operating on the rows first:

L	H	L	H
L	H	L	H
L	H	L	H
L	H	L	H

Operating on the columns we get:

LL	HL	LL	HL
LH	HL	LH	HL
LL	HL	LL	HL
LH	HL	LH	HL

Figure 3. Ordering of the wavelet coefficients under the lifted wavelet transform.

DETECTION AND LOCALIZATION

Template matching is a natural approach to pattern classification. In template matching, the goal is to locate/find in one or more target images that match a specific template. This is done by matching the template image to all (or many) of the possible places it could be located within the target image. A distance function (typically a simple Euclidean distance) is applied to the match to measure the similarity between the template and that location in the image. The algorithm then picks the location with the smallest Euclidean distance as the location of the object in the target image. Template matching works well when the variations within a class are due to additive noise. With other kinds of distortions like rotation or expansion, simple template matching does not work effectively.

Assuming that we have a pattern expressed as an $M \times N$ image $t(x, y)$, $x = 0, \dots, M-1$ and $y = 0, \dots, N-1$, and an $I \times J$ image $f(x, y)$, $x = 0, \dots, I-1$ and $y = 0, \dots, J-1$, where $M \leq I$, $N \leq J$. The goal is to find the best $M \times N$ block, which matches t within f . Overlaying the pattern r over the image t and shifting it to all positions can achieve this goal. For each of the positions, the difference between $r(x, y)$ and $t(x, y)$ is computed and according to

$$d_{f,t}^2(u, v) = \sum_{x,y} [f(x, y) - t(x-u, y-v)]^2 \quad (3.1)$$

The position that gives the minimum distance $d_{f,t}$ is the best match for r within t . The cross-correlation can be clarified by the expansion of d as follows:

$$d_{f,t}^2 = \sum_{x,y} [f^2(x, y) - 2f(x, y) \cdot t(x-u, y-v) + t^2(x-u, y-v)] \quad (3.2)$$

The term $\sum t^2(x-u, y-v)$ is constant. If the term $\sum f^2(x, y)$ is approximately constant, then the remaining cross correlation term

$$c(u, v) = \sum_{x,y} f(x, y)t(x-u, y-v) \quad (3.3)$$

is a measure of the similarity between the template and the $M \times N$ block in the image.

There are several disadvantages to using (3.3) for template matching: (1) if the energy varies with position, matching using (3.3) can fail, (2) the range of cross-correlation is dependent on the size of the pattern, and (3) the value of the cross correlation term is not invariant to changes in image amplitude such as those caused by changing lighting conditions across the image sequence.

Better results can be achieved by normalizing the image and the pattern and using the normalized cross-correlation as a measure of similarity.

$$\gamma(u, v) = \frac{\sum [f(x, y) - \bar{f}_{u,v}][t(x, y) - \bar{t}]}{\sqrt{\sum_{x,y} [f(x, y) - \bar{f}_{u,v}]^2 \sum_{x,y} [t(x-u, y-v) - \bar{t}]^2}} \quad (3.4)$$

Where \bar{t} is the mean of the template and $\bar{f}_{u,v}$ is the mean of $f(x, y)$ in the region under the template.

We consider the basic compression system illustrated in Figure 4. For the purpose of this work, we assume that only intra-frame wavelet based compression techniques are used analogous to motion JPEG2000. The transmitted data is the Wavelet coefficients and the goal is to detect and locate targets in the wavelet domain without calculating the inverse wavelet transform in order to reduce the computational complexity.

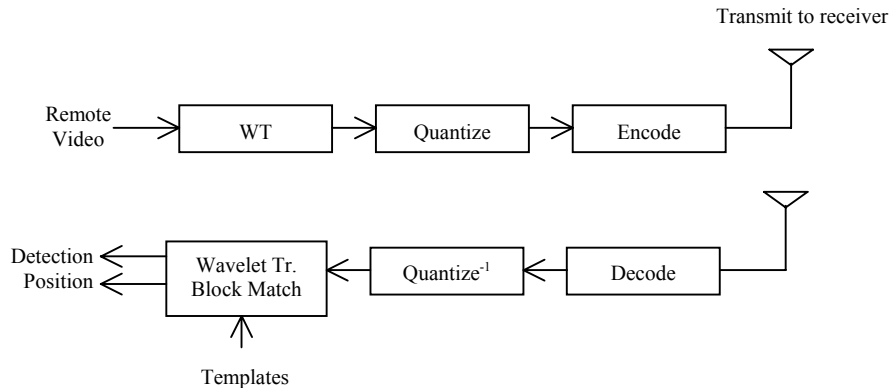


Figure 4. Basic wavelet based detection system setup. Video encoder/decoder and Wavelet-based detection.

The easiest way to detect and locate a target in the wavelet domain is, to use the template matching technique. Using only one template, the results are poor. These poor results are due to the fact that the wavelet-transformed blocks are shift varying by an amount related to the depth of wavelet decomposition used for the transform. For every single level of decomposition in the wavelet domain coefficients are shift variant by an additional factor of two. To counter this effect we use multiple templates formed by shifting the original template in different directions.

TEMPLATE DESIGN

By using additional templates, however, we can improve upon these results. The main idea behind multiple templates is to create insensitivity to different relative shifts. When designing additional templates, we consider two techniques: Full Size Average Fill and Reduced Windowing.

Full Size Average Fill (FSAF): In this case the first template is the target itself and the other templates are shifted versions of it. FSAF allows each template to retain the same dimensions as the original target: i.e., the maximum allowed size. Unfortunately a problem arises when building the shifted templates. Specifically, how do we fill in pixels uncovered by the shifts? In FSAF we fill these areas with predicted information. If we have enough information about the video sequence to guess what the neighborhood of the target looks like, this technique can give good results because of the positive effects of using the maximum template size. In Film #1, for example, the objects are widely separated and a desert background having a predictable gray level surrounds each object. In this case, we can add the needed information to build our shifted templates using this technique. The performance of this technique degrades if no accurate information is available about the target's neighborhood, however. Without such information, trying to guess the background levels does not help; rather, it completely corrupts our shifted templates. Fig. 5 explains how to build shifted templates using FSAF. In Fig. 5, the box with the solid-line sides represents the original target; and the box with the dotted-line sides represents a shifted template. Within the shifted template, the gray-dotted area is the extra area to be filled with predicted gray levels to build the template. Figure 5b illustrates the different shifts used to build up to 17 templates.

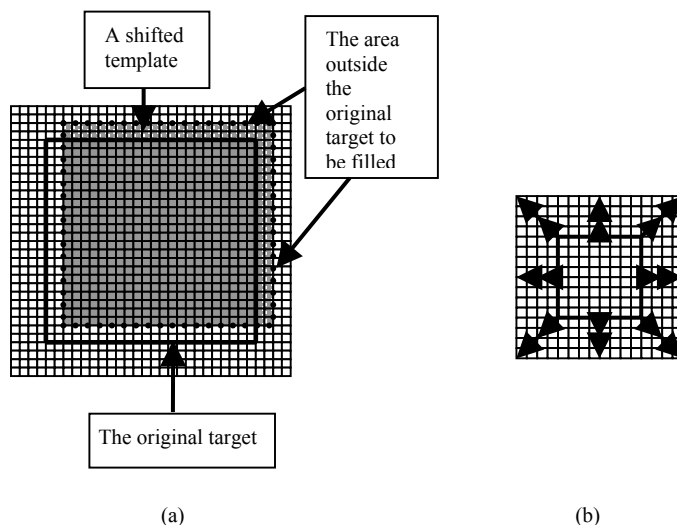


Figure 5. Creating shifted templates using FSAF. (a) Creating a shifted template, (b) Different shifts to construct 17 templates.

Reduced windowing (RW): If we do not have enough information to guess what the neighborhood of the target looks like, RW is the better choice. In this case, we have to sacrifice some of the information we have when building the shifted templates. For example, if our target is a 64×64 object, we have to create 32×32 -shifted templates within the 64×64 object. RW avoids the problems that might be caused by adding wrong information, but it also discards some of the available information. For small targets, RW does not work well. If the target is large enough, however losing a small amount of information will not significantly affect the total amount of information available for template creation. Fig. 6 explains building shifted templates using RW. In Fig. 6a, the solid-side box represents the target and the dotted-side box represents the first template. We can create additional template by shifting the first template within the target. Figure 6 b represents the shifts used to create 17 shifted templates.

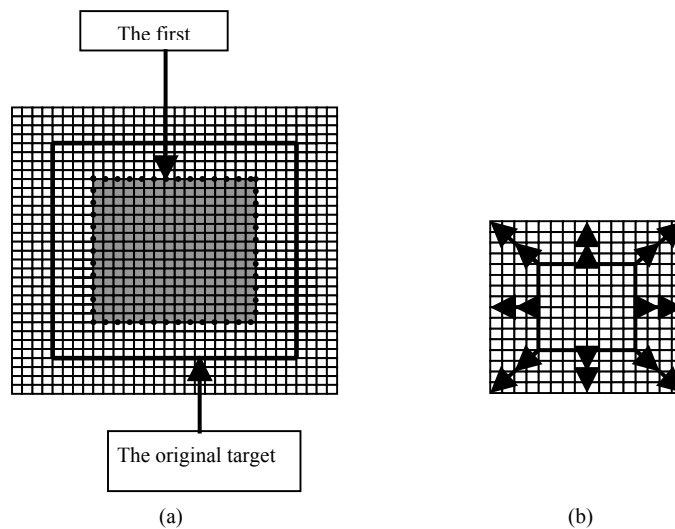


Figure 6. Creating shifted templates using RW (a) The original target and initial template, (b) Different shifts to construct 17 templates.

PERFORMANCE

The actual target coordinates are obtained by doing correlation match of the desired template to each frame of the given video sequence, then the correct coordinates are chosen after visual confirmation. The method used here to detect and locate a target in the wavelet domain is spatial correlation; the target is detected if the correlation between the template and the given target is high. Templates are actually obtained from a frame in the video sequence. Sample frames and sets of templates used are as shown in Fig. 7(a) and Fig. 7(b). Then the transform of the template and the transform of the frames are compared to get the wavelet based detection results.

The results presented here compare the following

1. Reduced windowing and Average fill.
2. Wavelet transforms $9/7, 5/3$ and the Haar or $(2,2)$.
3. Depth of wavelet decompositions.
4. Encoding bit-rate.

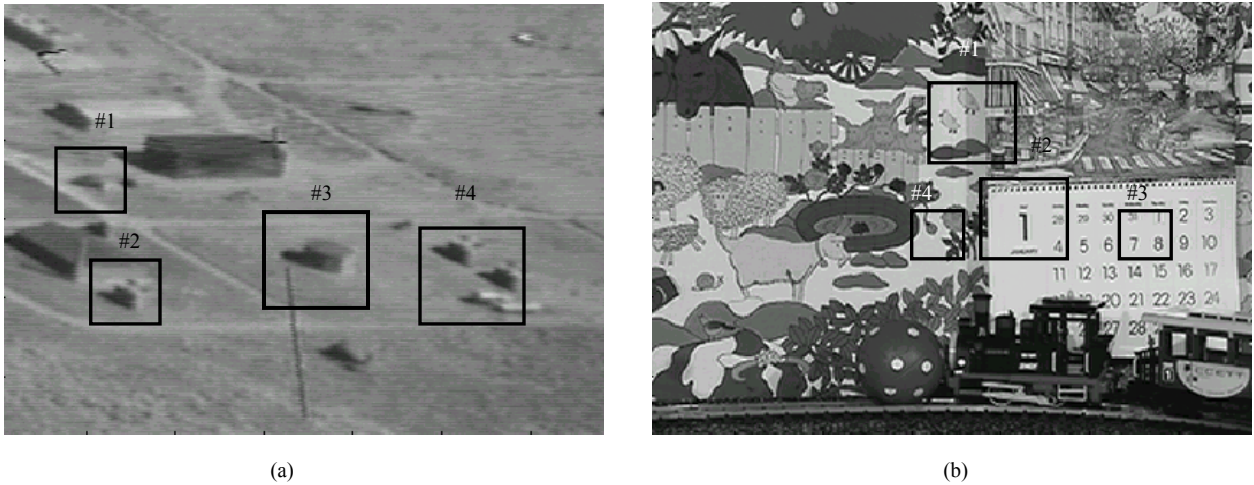


Figure 7. (a) Frame #25 of FHV_CIF Film #1. (b) Frame #114 of mobile_CIF Film #2. The different 32×32 and 64×64 templates used are indicated by the square blocks and are numbered 1 through 4 for each film.

Reduced Windowing (RW) Vs. Full Size Average Fill (FSAF): The wavelet transform of an image splits the image into different frequency sub-bands, at different rates. The two design techniques we explore here are the Reduced Windowing technique and the Full Size Average Fill. The full size average fill technique, obtains the shift by moving the template in a desired direction and filling the remaining portion of the template with an average gray scale value. The Reduced Windowing method chooses a template that is larger than the object, so that a template formed by shifting the required object does not lead to any loss of information.

Table 1 (Comparing RW and FSAF), Wavelet – 9/7, Wavelet depth of decomposition - 3, Film #1, Template #3 (64x64), 50 frames, Bit-rate -1bpp.

# Of shifted temp.	FSAF % Correct detection	RW % Correct detection
1	52	52
3	75	98
5	89	100

Table 1 above shows that the RW technique is much better than the FSAF. This is because, in the FSAF, method structure is lost when the new region formed by shifting the template is filled with a constant value. In the RW method, the whole object is preserved.

To maintain the comparability of results we assume that the RW gives the best detection compared to FSAF. Thus for all the following experiments only the RW method is used.

Wavelets 9/7,5/3 and the Haar: The most popular wavelets used for compression are, the 9/7 and 5/3 [6] [7]. Here we compare these different wavelet transforms with each other for single and multiple templates. The multiple template sets of 3 is created by shifting the template frame inside the object in the up down direction by four pixels, and that of 5 is created by adding left and right shifts to the 3 template set.

The following tables compare the different wavelets 9/7, 5/3 and the Haar, for different object sizes and different films. Tables 2 and 3 compare the different wavelets for a template size of 64×64, for films #1 and #2 respectively. The error tolerance is measured in Euclidean distance. Here the distance is considered to be 8 pixels.

Table 2 (Film #1 – template #3)

# Of shifted temp.	Wavelet 9/7 % Correct detection	Wavelet 5/3 % Correct detection	Wavelet Haar % Correct detection
1	72	83	86
3	98	100	100
5	100	100	100

Table 3 (Film #2 – template #1)

# Of shifted temp.	Wavelet 9/7 % Correct detection	Wavelet 5/3 % Correct detection	Wavelet Haar % Correct detection
1	73	62	79
3	88	87	90
5	100	100	100

Table 4 and 5 compare the different wavelets for a template size of 32×32 for films #1 and #2 respectively.

Table 4 (Film #1 – template #4)

# Of shifted temp.	Wavelet 9/7 % Correct detection	Wavelet 5/3 % Correct detection	Wavelet Haar % Correct detection
1	78	77	79
3	92	89	95
5	100	100	100

Table 5 (Film #2 – template #4)

# Of shifted temp.	Wavelet 9/7 % Correct detection	Wavelet 5/3 % Correct detection	Wavelet Haar % Correct detection
1	28	32	35
3	47	52	57
5	72	72	76

From Tables 2 through 5 we infer that the Haar wavelet gives a slightly better result than both the 5/3 and 9/7 wavelets. The 9/7 and the 5/3 wavelets give similar results. From Table 5 we see that the results are poor for template #4 from film #1 and that it improves with the number of shifted templates.

Depth of wavelet decomposition: The wavelet transform decomposes an image into its different frequency subbands. A depth N decomposition of an image means that the low frequency components of the image are successively transformed N times. The lifted wavelet transform stores the pixel coefficients corresponding to the original position of the pixels in the image, but there is a shift variance introduced. The period of the shift variance increases with the number of times the image is passed through the wavelet transform. This can lead to bad detection results, thus we improve upon the results by using a large set of templates formed by shifting the image in different

directions. Here the templates are shifted up, down, left and right and in diagonal directions i.e. up-left, up-right, down-left and down-right by 4 or 8 pixels in each direction.

Table 6 (Table showing detection results for depth of decomposition of 4.)

# Of shifted temp.	Object # 3 Film #1 % Of correct detection	Object #1 Film #2 % Of correct detection
1	38	22
3	62	45
5	69	47
9	83	72
13	89	75
17	95	89
21	99	95

Table 6 shows the correlation results for 64×64 objects from both film #1 and film #2. We can infer, by comparing results obtained from Table 2 and Table 6, that the correlation values are much better for low depth of decomposition. The detection results can be improved by increasing the number of shifted templates.

Encoding Bit-Rate: The bit-rate of the encoder determines the quality of the wavelet coefficients arriving at the receiver. Low bit rates mean loss of information. Since correlation depends on structure of the given image, any loss of information will be detrimental to the detection results.

Table 7 (Film #1 – Template #3)

Bit Rates (bpp)	Wavelet 9/7 % Correct detection	Wavelet 5/3 % Correct detection	Wavelet Haar % Correct detection
1	100	100	100
0.5	100	100	100
0.025	100	100	100
0.015	88	88	100
0.005	56	54	57
0.0005	5	9	6

Table 7 describes the detection performance for different encoding bit rates. The template chosen for the detection was template #3 of size 64×64 from film #1. When the bit rate is reduced below a certain level the detection results are very poor. In the above case the performance is good for bit rates greater than 0.015 bits per pixel (bpp) i.e. the encoder does a good job of preserving the coefficients of the wavelet transform for bit rates above 0.015 bpp.

CONCLUSION

The above results indicate that it is possible to do detection in the wavelet domain, and obtain good results. We note that the quality of detection is directly proportional to the number of shifted templates. Even if the results are not good, as in the case for depth of decomposition of 4, we can improve the detection by increasing the number of shifted templates. Results from table 7 indicate that the detection is unaffected by the decrease of bit-rates as low as 0.015 bpp.

In the future, we need to study in detail the application of correlation in the different subbands of the wavelet transformed image, and come up with a model based on it that can improve detection performance. Some templates do not give as good detection results as others as shown by the case in Table 5. So a study of these specific templates can help in determining what sort of templates are better suited for doing wavelet based detection.

REFERENCES

- [1] W. Brent Seales, Cheng J. Yuan, Wei Hu and Matthew D. Cutts, "Object recognition in compressed imagery," *Image and Vision Computing* 16(1998) 337-352
- [2] David S. Taubman and Micahel W. Marcellin, "JPEG 2000 Image compression fundamentals and practices," Kluwer Academic Publishers, Boston, Ma. 2002.
- [3] Amir Said, and William A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchial trees," *IEEE transactions on Circuits and Systems for Video Technology*, Vol. 6. No. 3, June 1996.
- [4] Rafael C. Gonzalez and Richard E. Woods, "Digital Image Processing," Second Edition, Prentice Hall, Upper Saddle River, NJ 07458 pp 349-404
- [5] Wim Sweldens, "Wavelets and the Lifting Scheme: A five Minute Tour," *Zeitschrift für Angewandte Mathematik und Mechanik*, vol. 76 (Suppl. 2), pp. 41-44, 1996.
- [6] A. Cohen, I. Daubechies, and J.C. Feauveau, "Biorthogonal bases for compactly supported wavelets," *Commun. Pure Appl. Math.*, pp. 485-560, Apr. 1995.
- [7] I. Daubechies, "Ten Lectures on Wavelets," Society for industrial and applied mathematics, Philadelphia, Pennsylvania 1992 pp. 278-287.