PROCEEDINGS OF SPIE

SPIEDigitalLibrary.org/conference-proceedings-of-spie

# Feature-based correlation filters for distortion invariance

Samuel Peter Kozaitis
Robert Petrilak
Wesley E. Foor

# Feature-based correlation filters for distortion invariance

Samuel P. Kozaitis, R. Petrilak

Florida Institute of Technology,
Department of Electrical and Computer Engineering
Melbourne, FL 32901-6988

W. E. Foor

Rome Laboratory,
Photonics Center
Griffiss AFB, NY 13441

## ABSTRACT

In an optical correlator, binary phase-only filters (BPOFs) that recognize objects that vary in a nonrepeatable way are essential for recognizing objects from actual sensors. An approach is required that is as descriptive as a BPOF yet robust to object and background variations of an unknown or nonrepeatable type. We developed a BPOF that was more robust than a synthetic discriminant function (SDF) filter. This was done by creating a filter that retained the invariant features of a training set. By simulation, our feature-based filter offered a range of performance by setting a parameter to different values. As the value of the parameter was changed, correlation peaks within the training set became more consistent and broader. In addition, the feature-based filter was potentially useful for recognizing objects outside the training set. Furthermore, the feature-based filter was more easily calculated and trained than an SDF filter.

## 1. INTRODUCTION

The major difficulty encountered when using a binary phase-only filter (BPOF) in an optical correlator is its sensitivity to changes in the object's appearance. In imagery from actual sensors, the same object can vary significantly depending on aspect angle, lighting, atmospheric effects, and a host of other variables. In addition, object boundaries may be poorly defined and buried in the background. Identifying an object that has a nonrepeatable signature is one of the key technical challenges of automatic object recognition.[1]

In reality, imagery from sensors is often nonrepeatable; the same object can vary in appearance depending on weather conditions, lighting, and other variables. These objects, whose distortion cannot be repeated exactly, are said to vary in an unknown manner. The creation of BPOFs to recognize objects that vary in an unknown manner is essential for recognizing objects from actual sensors. Recently, approaches based on SDF filters have been applied to areas such as handwriting verification, and aircraft identification.[2-3] In one study, many variations of an pattern was incorporated into the filter to obtain robustness.[2] Another approach used digital image processing techniques to confine images to vary in a limited but unknown manner, then filters with a small number of training images were used to identify objects.[3] In both cases, an approach is required that is as descriptive as a BPOF yet robust to object or background variations of an unknown or nonrepeatable type.

 0-8194-0866-2/92/$4.00

We derived a filter that was calculated from object's' features. By attempting to recognize an object based on its features, we made a BPOF more robust. To help evaluate the potential of our approach, we used imagery from actual sensors that were not from the original training set. The next section briefly discusses thresholding and is followed by SDF filter formation in the following section. Details of a filter made from features of a training set are in Section 4, and Section 5 presents the performance of the filter. Finally, discussion and conclusions are presented in Sections 6 and 7 respectively.

## 2 PREPROCESSING

Infrared (IR) imagery (8-14 μm) of ground scenes from actual sensors were used to evaluate filters. Images were digitized with 128 x 128 pixels with 8 bits/pixel. Because the application was to binary SLMs, the imagery ultimately had to be thresholded.

Thresholding was performed by choosing a single threshold value for the entire image. When the object and background within an image are obvious, a threshold value can be easily chosen, and different methods usually give similar results. In the imagery we examined, the background and object were easily separated; however, edges were not well-defined. We used digital image processing techniques to implement an isodata thresholding method that examines peak values in the histogram of an image.[4] A threshold value was chosen between peaks that were associated with the object and the background so that the object could be segmented from the background. If noise or atmospheric distortion is present, the peaks of the histogram will change their position or shape, but the peaks are usually identified. Choosing a threshold value between peaks of a histogram often results in an image that is similar to the silhouette of the object. As variables such as lighting, noise, and atmospheric effects within an image change, the resulting thresholded images will remain similar but will often be different in an unknown or nonrepeatable way.

## 3 SYNTHETIC DISCRIMINANT FUNCTION FILTERS

We compared our filter to results obtained with an fSDF filter, so we briefly describe how the fSDF filters were made. A conventional SDF is a combination of images that can be described as

$$s(x,y) = \sum_{n} a_n t_n(x,y) \tag{1}$$

where $t_n$ are centered training images and $a_n$ are weight coefficients. SDF synthesis techniques may be used to determine the weight coefficients.[5,6] The complex conjugate of the Fourier transform of $s(x,y)$ is the matched filter

$$S(u,v) = F[s(x,y)]^* \tag{2}$$

where $F$ is the Fourier transform operator.

An improved version of an SDF, called a filter SDF (fSDF), has been introduced that includes the function modulation characteristics of the device onto which the filter is mapped in the synthesis equations.[7] For fSDF-BPOFs, the coefficients an can be iterated based on the formula

$$a_n^{i+1} = a_n^i + \beta\left[c_n - c_0\left(\frac{m_n^i}{m_0^i}\right)\right] \tag{3}$$

where $i$ is the iteration number, $\beta$ is a damping constant, and $m^i_n$ is the modulus of the peak correlation response of image $t_n(x, y)$ with a filter made with the coefficient vector $\boldsymbol{a}^i$. In the experiments described in this paper, the initial solution vector was taken to be the desired correlation response vector, $\boldsymbol{a}^0 = \boldsymbol{c} = 1$.

## 4.0 FEATURE-BASED FILTERING

Generally, the cross-correlation between a input image and an image used to make a filter is maximized when the mean squared error (MSE) between the two images is minimized.[8] In the formulation of an SDF filter, the MSE between each training image and the filter image are minimized subject to the constraint that all the MSEs are equal. This is accomplished by multiplying each input image by a weight value. The idea behind the fSDF filter is similar to that of the SDF filter; however, the weights are chosen to produce the desired filter after the SDF has been quantized to form a BPOF.

In contrast to the SDF filter formulation, we developed a filter whose values were determined by features of the training set. We attempted to find a filter that represents the critical characteristics of an object so that objects outside the training set could be identified. Thereore, we examined features that were invariant with respect to the training set. Because correlation filters are derived from the Fourier transform of an object, we examined feature extraction in the Fourier domain. Optical feature extraction has been previously performed in the Fourier domain by using the power spectrum of an object as features to train a neural network for classification.[9] Because BPOFs have provided suitable solutions for recognizing objects, we viewed the BPOFs from several objects as a collection of features to create a new filter for an optical correlator. We retained those features that were invariant among a training set by examining all filters at the same pixel on a pixel by pixel basis. We rejected those features that were not invariant for a particular distortion. For example, if the same value occurred in the BPOFs at a particular pixel, we used that value for that pixel in a new BPOF. In this way we constructed a new BPOF that represented features of the objects in the training set.

Our filters were generated from $n$, $N \times N$ training images as follows. The BPOF for each training image was calculated. Then, the BPOFs of all the training images are added together pixel by pixel. The result was an $N \times N$ array of integer values. If the value at a pixel of the array was greater than a positive-valued threshold, then that pixel was set to +1. If the value at a pixel of the array was less than the negative of the threshold, then the pixel was set to -1, otherwise the pixel was set to zero. The filter was described as

$$G(u,v) = \begin{cases} 1 \ if \ p \le \sum_n H_n(u,v) \\ -1 \ if (-p) \ge \sum_n H_n(u,v) \\ 0 \ otherwise \end{cases} \tag{4}$$

where $H_n(u, v)$ is the BPOF of a training image $t_n(x, y)$ and $p$ is the threshold that determines the order of the filter.

## 5.0 EXPERIMENT

We considered an IR photograph like that described in the previous section that contained a total of ten aircraft. We created ten 128 x 128 images from the photograph where each aircraft could fit in a square of about 90 x 90 pixels and labeled the images image1 - image10. We used five thresholded images that were derived from Fig. 1 as training images to create different filters. We also obtained nine additional images (image11 - image19) from another photograph that contained the same type of aircraft. We then cross-correlated both filters with the training set and the remaining fourteen images. Both the fSDF and feature-based filter with $p = 1$ had similar performance. For images within the training set, the average correlation height produced by the fSDF and feature-based filter were 70.1 and 76.4% the maximum of that filter respectively. The maximum value of the feature-based filter was 90.8% of the maximum of the fSDF filter. The PRMSR values for images within the training set were 11.6 for the feature-based filter compared to the 15.3 for the fSDF filter. Neither filter produced very useful correlation heights for images outside the training set. The feature-based filter produced more consistent results than the fSDF filter at the expense of slightly decreased and broader correlation peaks.

A comparison of the normalized correlation heights between the fSDF filter and feature-based filter with $p = 3, 5$ is shown in Fig. 2. The maximum value for each filter was set to 100 as in the previous experiment. For images within the training set, the average correlation height produced by the feature-based filters for $p = 3, 5$ were 78.3 and 94.5% the maximum respectively and the PRMSR values were 9.6 and 3.2 respectively. For $p = 3$, the correlation peaks were slightly broader than the $p = 1$ case; however, the $p = 5$ filter had significantly broader peaks. The maximum values of the correlation height for $p = 3, 5$ were 52.8 and 11.1% of the maximum of the fSDF filter respectively.

In contrast to the fSDF filter, some correlation responses for images outside the training set were useful using a feature-based filter with $p =3, 5$. For the feature-based filter with $p = 3$, some correlation heights were above 30% the maximum. For $p = 5$, some values were above 60%. As in the previous experiment, the feature-based filter produced more consistent results than the fSDF filter at the expense of decreased and broader correlation peaks.

The correlation peak for the feature-based filter when $p = 5$ was broad when compared to other filters with different values for $p$ or the fSDF filter. In an effort to improve the PRMSR values, we considered blocking pixels of the filter near the DC component. The blocking of the DC is consistent with our approach because all objects will have a DC component; therefore, this feature will not be useful for discrimination and can be eliminated. We blocked the central pixel of the filter where $p = 5$ and compared its results to the case where the central pixel was not blocked. The normalized correlation heights are almost the same in both cases but the PRMSR values had increased from 3.2 to 4.2. We also blocked a 3 x 3 pixel area of the filter around the DC component and the PRMSR further increased to 5.3 for the training set but the average normalized correlation heights generally decreased. A summary of the performance of the various filters are shown in Table 1. As fewer but more consistent features were retained, the correlation responses were more consistent but the correlation response became broader. By blocking the pixels around the DC component, the correlation height could be made sharper at the expense of less consistent correlation heights. As an example of the correlation results obtained with the different filters, the cross-correlation responses for image4 and the feature-based filters are shown in Fig. 3 for comparison. Note that the height of each response was set to the same value and Fig. 3(c) was for the case when the central pixel of the filter was blocked.

**Table 1: Performance of Various Filters for Images in Training Set**

| Filter | Average Normalized Correlation Height | Average PRMSR value |
|---|---|---|
| fSDF | 70.8 | 15.3 |
| Feature-based (p=1) | 76.4 | 11.6 |
| Feature-based (p=3) | 78.3 | 9.6 |
| Feature-based (p=5) | 94.5 | 3.2 |
| Feature-based (p=5) 1 pixel DC block | 94.0 | 4.2 |
| Feature-based (p=5) 3x3 pixel DC block | 87.8 | 5.3 |

In contrast to the fSDF filter, the feature-based filters were potentially useful for images outside the training set. We arbitrarily set a threshold in the correlation plane as a percentage of the maximum correlation height of the training set to examine the responses of the different filters. A correlation response greater than the threshold indicates recognition of an object. Although changing the threshold would change the results, similar conclusions can be drawn unless the threshold value is too low or too high. We examined the case where the threshold is 30% of the maximum and examined the data for the feature-based filters for correlation results outside the training set. The data was summarized in Table 2 and showed that the performance of the fSDF filter and the filters with $p = 1, 3$, were similar. These filters were generally not useful for recognizing images outside the training set; a total of 1 of the 14 images outside the training set was recognized for these filters. When $p$ was increased to 5, 12 of the 14 images were recognized. As pixels near the DC were blocked to improve the PRMSR value, the average correlation heights of the responses decreased. Therefore, fewer images could be recognized. The performance of the filters with $p = 5$ with and without the central pixel blocked were similar.

**Table 2: Performance of Various Filters for Images outside of Training Set**

| Filter | Number of images recognized (above 30% of maximum) | Average Normalized Correlation Height of recognized images |
|---|---|---|
| fSDF | 0 | - |
| Feature-based (p=1) | 0 | - |
| Feature-based (p=3) | 1 | 34.6 |
| Feature-based (p=5) | 12 | 47.6 |
| Feature-based (p=5) 1 pixel DC block | 12 | 45.4 |
| Feature-based (p=5) 3 x 3 pixel DC block | 7 | 42.2 |

## 6.0 DISCUSSION

The feature-based filter recognized a set of objects based on features. This was done by creating a filter that retained features that were invariant with respect to the training set. The feature domain contained the sign of the phase of the Fourier transform of training images at each pixel. By using features that were invariant to the training set, objects that were outside the training set could be recognized.

The feature-based filter was calculated in a simpler manner than the fSDF filter. Both the time and number of operations required to calculate the feature-based filter was on the order of calculating $n$ FFTs. In contrast, the fSDF approach requires a cross-correlation between every training image and the filter every iteration. This requires on the order of $ni$ FFT calculations where $i$ is the number of iterations and has been generally set to ten.[7] Therefore, the time and number of operations required to calculate the feature-based filter was about an order of magnitude less than for the fSDF filter.

A feature-based filter was more easily trained than an fSDF filter. We consider the case where an fSDF filter made from $n$ training images is presented with a new image that belong to the same class as the training set but does not produce a useful correlation result. In this case, the fSDF filter must be completely recalculated in the usual way with $n + 1$ images in the training set. For $k$ new images, $(n+k)i$ FFTs are required to generate a new filter. In contrast, the feature-based filter required $k$ FFTs when $k$ new images were presented.

## 7.0 CONCLUSION

The feature-based filter offered a range of performance. In the case where none of the pixels were set to zero in the filter, the fSDF and feature-based filter offered similar performance. The feature-based filter was slightly more consistent and had broader correlation peaks for objects within the training set than the fSDF filter. Neither filter appeared to be useful for recognizing objects outside the training set.

As pixels of the filter were set to zero in the feature-based filter, the correlation peaks within the training set became more consistent even though their average height decreased. As the number of pixels set to zero increased, the correlation heights became more consistent but broader. When images of the same class as the training set but not in the training set were used as inputs, the feature-based filter was potentially useful. Our experiments involved five training images. The use of more training images suggests that more possibilities are available in trading off between consistency and broadness of the correlation results. In this way, the feature-based filter can be made robust to recognized object outside the training set.

Finally, a feature-based filter was more easily calculated and trained than an fSDF filter. The feature-based filter required about an order of magnitude fewer calculation than the fSDF filter. In addition, new images added to the training set, required the fSDF filter to be completely recalculated which was not the case in the feature-based filter.

## 8.0 REFERENCES


[1] M. W. Roth, "Survey of neural network technology for automatic target recognition," *IEEE Trans. Neur. Net.* 1(1) 28-43 (1990)

[2] T. S. Wilkinson, D. A. Pender, and J. W. Goodman, "Use of synthetic discriminant functions for handwritten-signature verification," *Appl. Opt.* 30(23), 3345-3353 (1991)

[3] S. P. Kozaitis and W. E. Foor, "Performance of synthetic discriminant functions for binary phase-only filtering of thresholded imagery," *Opt. Eng.* 31(4) (1992)

[4] R. Kohler, "A segmentation system based on thresholding," *Comp. Vision, Graphics & Im. Proc.*, 15(4), 319-338 (1981)

[5] D. Casasent, "Unified synthetic discriminant function computational formulation," Appl. Opt. 23(10), 1620-1627 (1984)

[6] Z. Bahri and B. V. K. Vijaya Kumar, "Generalized synthetic discriminant functions," *J. Opt. Soc. Amer. A* 5(4), 562-571 (1988)

[7] D. A. Jared and D. J. Ennis, "Inclusion of filter modulation in synthetic discriminant function construction," *Appl. Opt.* 28(2), 232-239 (1989)

[8] P. Maragos, "Morphological correlation and mean absolute error criteria," *Proc. of ICASSP* '89, IEEE, 1568-1571 (1989)

[9] D. Clark and D. Casasent, "Practical optical Fourier analysis for high speed inspection," *Opt. Eng.* 27(5), 365-371 (1988)

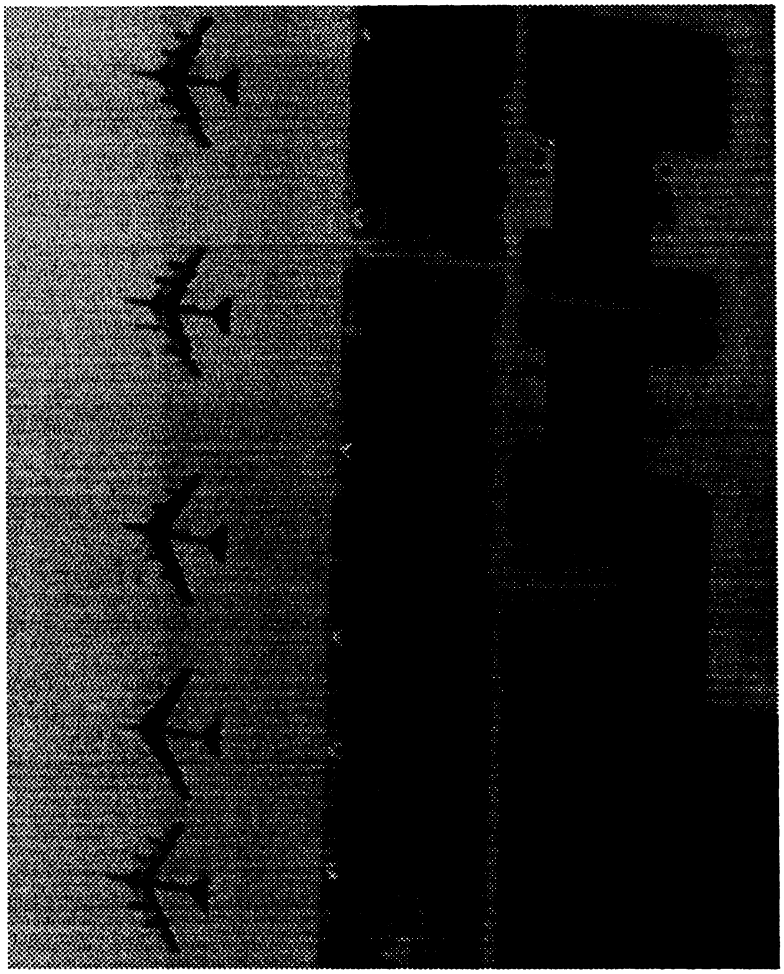


**Figure 1** Portion of IR image showing some of the images used in our experiment. Thresholded versions of the five aircraft were used as training images for different filters.

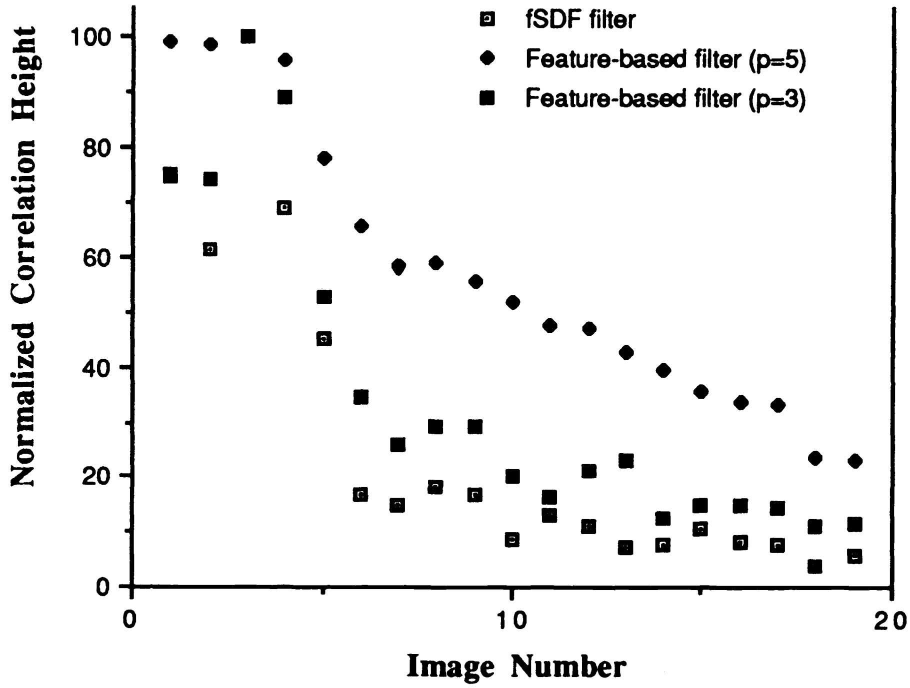



**Figure 2** Comparison of correlation heights obtained with feature-based filters with $p$ =3, 5, and an fSDF filter for various imagery. Note that only image1 - image5 are in the training set.

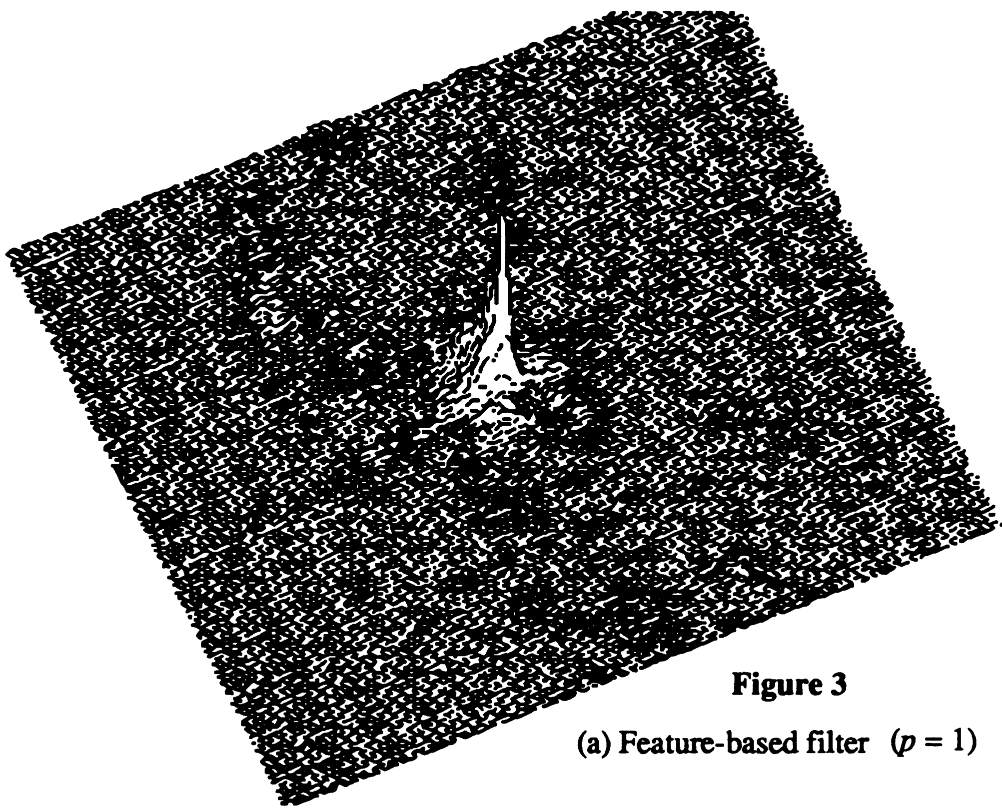
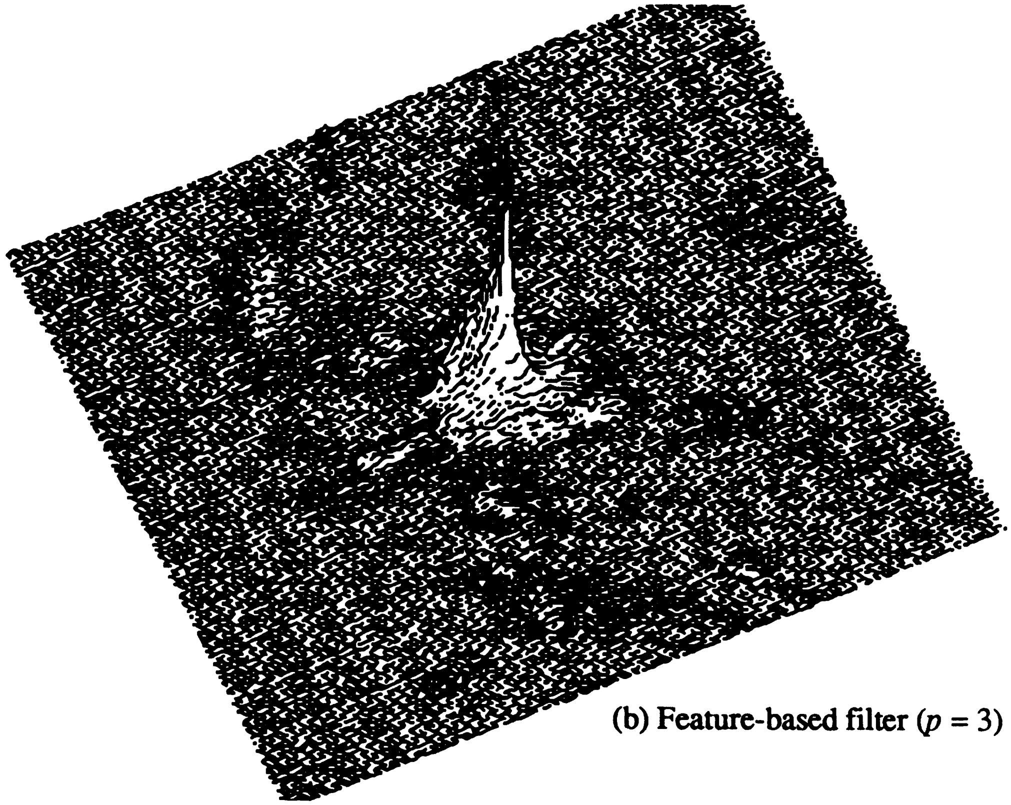


**Figure 3**

(a) Feature-based filter ($p$ = 1)



(b) Feature-based filter ($p = 3$)

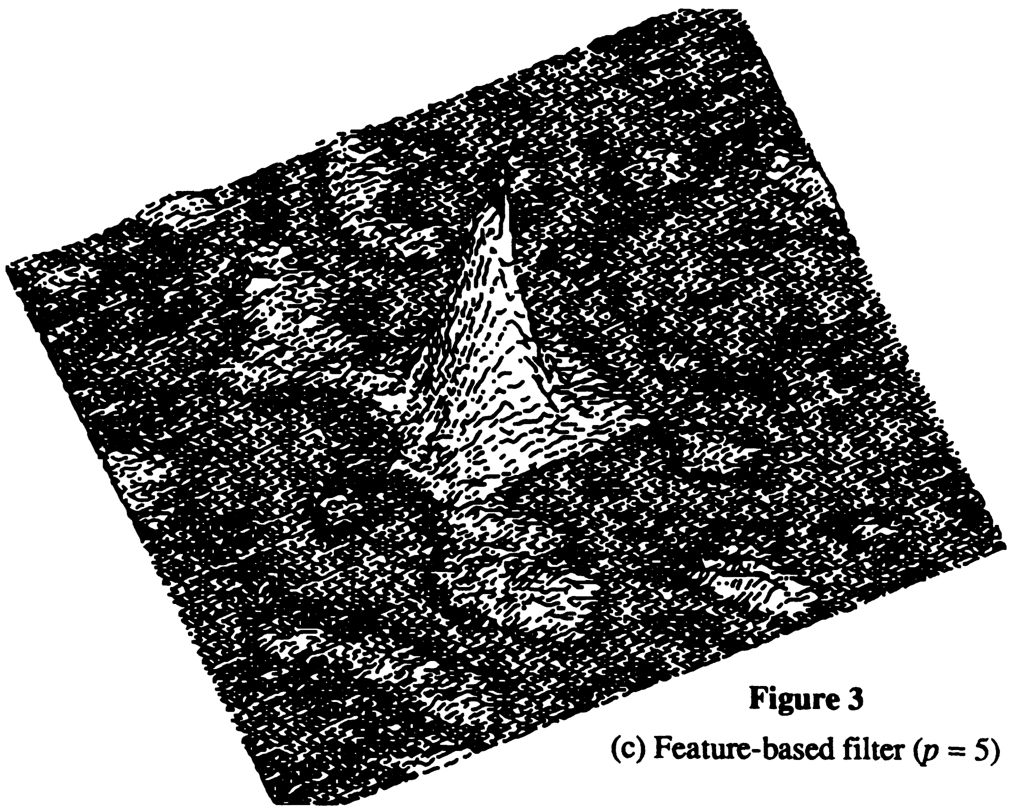


**Figure 3**

(c) Feature-based filter ($p = 5$)