CHAPTER 4

DETECTION OF PARTICLE SHAPES FROM THEIR IN-LINE HOLOGRAMS: NON-LINEAR POST-PROCESSING

4.1. Introduction

In-line holography is extensively used to analyze particle fields. One of the important features of particles is their shape. Cross-sectional shapes of particles can be decoded from their in-line holograms by incorporating the digital reconstruction technique given in previous chapters with some simple non-linear techniques.

The reconstruction algorithm which suppresses twin-image is a rather general purpose algorithm, since the only assumption about the object distribution, $a(x,y)$, is that it is real. As mentioned before, it is possible to achieve better reconstructions if more restrictions are imposed on the object distribution. The general purpose linear reconstruction algorithm of Chapter 3 can be either coupled or cascaded with another process (possibly non-linear) to achieve some special results. For instance, if it is known *a priori* that the object distribution function consists of particles with well defined sharp borders on a transperant background, this information can be utilized for reconstruction.

Most of the non-linear filters shows a phenomenon called the threshold effect: the performance of the filter detoriates quickly beyond a specific noise-to-signal ratio. Therefore, the reduction of noise prior to any non-linear process is essential for overall success. Because of this reason, the twin-image suppression, as given in Chapter 3, is very important to get useful results from any non-linear post-processing. In fact, the twin-image suppression is so succesful that even most simple non-linear processes such as thresholding at a predetermined level can be enough for complete recovery of input objects. Results of this very simple non-linear operation applied to both conventional and twin-image suppressed reconstructions are shown for comparison.

Another non-linear post-processing which is applied to the reconstructions of previous chapters is an edge detecting filter, namely a "covariance filter". The covariance filter is a very simple filter which requires modest computation. It was emphasized in previous chapters that holography (both recording and reconstruction) is an edge preserving operation in the ideal case. Since the simulations using synthesized objects are done according to the perfect mathematical structure, the reconstructed objects (either conventional or with twin-image suppression) have perfectly sharp edges. However, due to the low-pass nature of optical hologram recording, the sharp edges of the input objects are somewhat lost. The effect of edge softening is more important for smaller particles. Since the edge-detecting covariance filter gives good performance in the presence of well defined sharp edges, it may not be suitable for reconstructions from optical holograms directly. If this is the case, then a thresholding can be used to sharpen the edges. As indicated above, due to the reduced background as a consequence of twin-image suppression, thresholding at a predetermined level gives good results. No attempt is made in this dissertation to model the low-pass nature of optical hologram recording, and to compensate its undesirable effects based on its model.

The filtering system, which consists of a linear twin-image supression part (Chapter 3) and a non-linear shape decoding part (Chapter 4) is a complete filtering system which transforms an in-line hologram into particle shapes. The results of the filtering system can be readily used for further artificial intelligence type of image analysis, for such tasks as particle classification, counting, size measurements, location analysis, etc.

4.2. Thresholding

Quantization is one of the simplest non-linear procedures, which is used if it is known *a priori* that the original image consists of only a few levels of gray. Simply, the gray level of the image being processed is assigned to the closest allowable gray level. If the original image is known to be binary (i.e. if it consists of perfect white or perfect black areas), then a threshold is chosen, and the gray levels below the threshold are assigned to black and the others to white. This operation is called thresholding. One of the problems of thresholding is that the level of the threshold is unknown, and different for different images. In practice, a histogram of the image is computed to see dominant peaks and a threshold is chosen at a suitable level between the peaks. But this procedure is inconvenient for any automated application since it requires frequent operator intervention to select the suitable level of the threshold. Worse than that, if the noise level is high, then there may not be a threshold level which can perform the perfect separation of two original gray levels.

The objects used for holograms in this dissertation are mostly totally opaque particles on transparent background. Therefore, they appear as black small objects on white. Since this is known a priori, the reconstructed images from a hologram can be thresholded to recover the original particles. Also, since the particles occupy a small area of the overall image, a histogram of perfect reconstruction is expected to have one very sharp peak corresponding to the background white, and a small sharp peak for the dark particles. Particle area could be very small compared to overall picture area, and the small peak corresponding to dark particles may not be noticable, because of the scale used for the histogram. The reconstructions from holograms are never perfect and a variation of gray levels is expected. In the conventional digital reconstruction of optical holograms, the background consists of random noise and twin-image, therefore the histogram will have a thicker peak at the white region. (See Figs.46,47.) Again, since the dark area corresponding

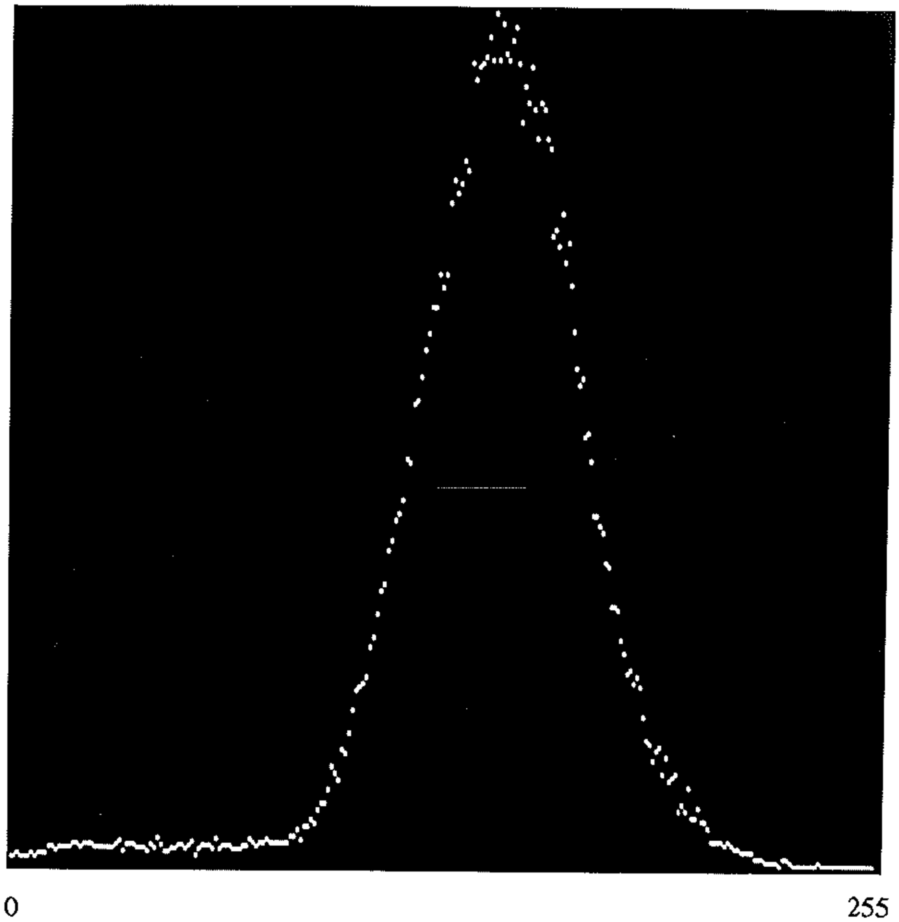

Figure 46. A typical histogram for conventional reconstructions.

to the particles is a small fraction, the peak corresponding to it may not be visible. Furthermore, because of the low-pass nature of the physical hologram recording set-up, the edges of the particles are smeared, and this results in a variation of gray levels corresponding to them. Thus, the threshold must be chosen (see Figs.46,47.) to assign the gray levels of particles to dark. But this means a higher threshold level. A higher threshold level may include the tail of the peak corresponding to the white background. If this is the case, then the result is disasterous, and it happens for most conventional reconstructions. Fortunately, the reconstruction with twin-image suppression decreases the background variation to the level of random noise, and a larger safe region for threshold is available due to the thinner peak. Furthermore, it is observed that the background noise level of this kind of reconstructions are consistently low enough to allow a fixed threshold to be used independent of the input hologram. (The twin image is more effective than random noise.) Thus there is no need to check the histogram of every reconstruction, instead a fixed level can be used satisfactorily. Therefore, the twin-image suppression as given in Chapter 3 decreases the background variations to a level which is good enough even for very simple non-linear techniques.

The results shown in Fig.48 are typical for conventional digital reconstructions and reconstructions with twin-image suppresion. Note that thresholding applied to the reconstructions with twin-image suppression gives results that are ready for many different image analysis operations such as measurement of size distributions and particle classification.

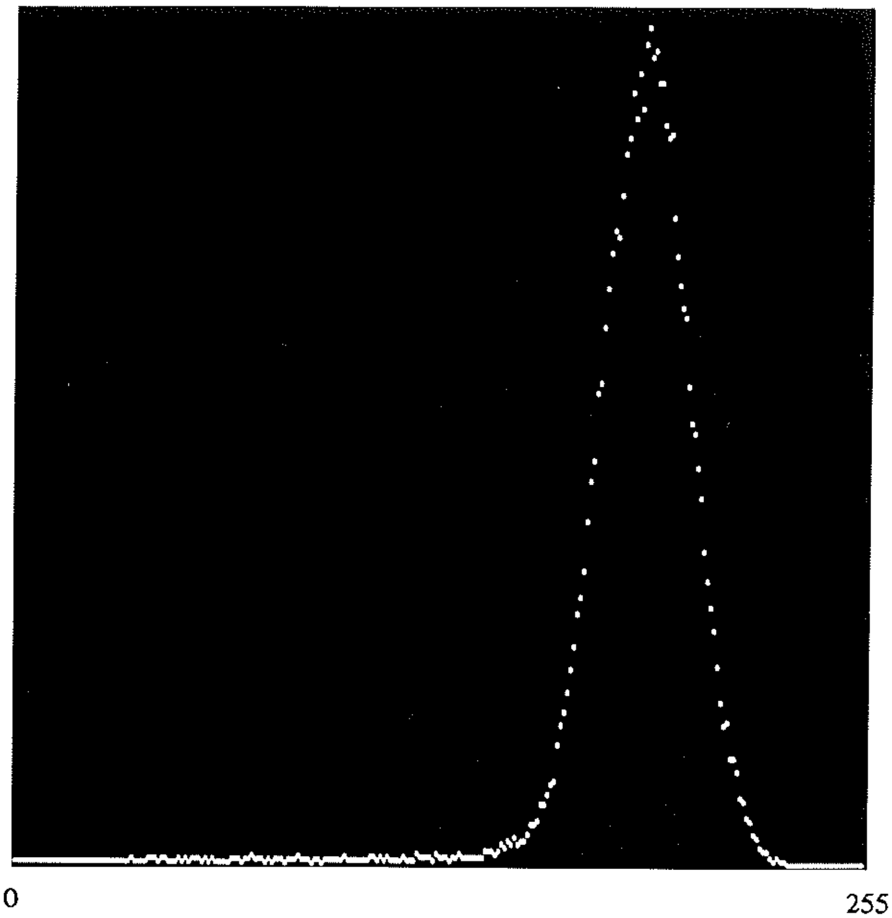

Figure 47. A typical histogram for reconstructions with twin-image suppression.

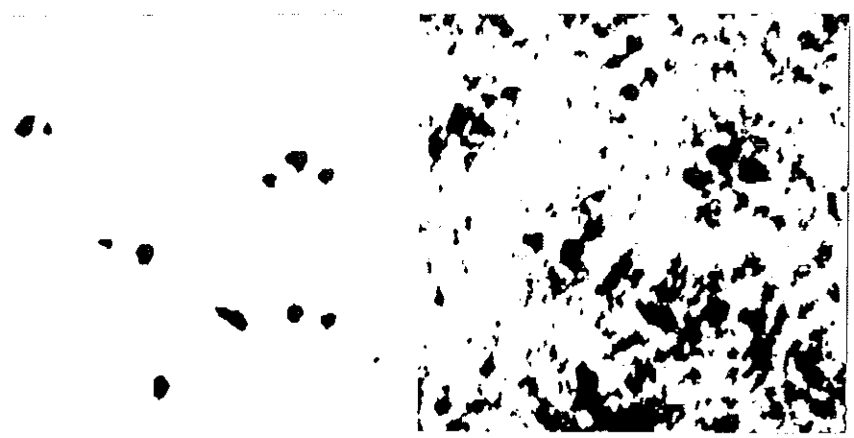

Figure 48. Effect of thresholding to a) reconstruction with twin-image elimination (left), b) conventional reconstruction (right).

4.3. The Covariance Filter

Many linear and non-linear filters for edge-detection can be found in the literature [49,50,53]. Some of these filters emphasize simplicity for the sake of easy implementation. On the other hand, some of them are based on statistical decision making and are quite sophisticated. The covariance filter presented in this chapter is a simple filter both conceptually and computationally. It is similar to Halé's second operator [50].

A small window of size $(2N+1) \times (2N+1)$ slides all over the image which is going to be filtered. At every location the covariance of the points within the window is estimated using,

$$s^2 = \frac{1}{(2N+1)\times(2N+1)} \sum_{m=-N}^{N} \sum_{n=-N}^{N} \left[p(m,n) - \bar{p}\right]^2 , \tag{4.1}$$

where, $p(m,n)$ is the pixel located at (m,n) with respect to the center pixel of the window (also called the location of the window). $\bar{p}$ is the average value of the pixels within the window. The covariance s^2, is the output of the filter, i.e., the pixel of the output image corresponding to the center of the window is replaced by s^2. Thus,

$$\begin{aligned} b(k,l) &= \frac{1}{(2N+1)\times(2N+1)} \sum_{m=-N}^{N} \sum_{n=-N}^{N} \left[a(k-m,l-n) - \bar{p}_{k,l}\right]^2 \\ \bar{p}_{k,l} &= \frac{1}{(2N+1)\times(2N+1)} \sum_{m=-N}^{N} \sum_{n=-N}^{N} a(k-m,l-n) . \end{aligned} \tag{4.2}$$

$a(k,l)$ and $b(k,l)$ denote the input and output images, respectively. If the input image size is $M \times M$, then the output image size is $M-N \times M-N$.

The idea behind the covariance filter is that if there is an edge within the window, then there will be a large variation of gray levels. Thus, an edge is characterized by a high covariance. If there is no edge, then the covariance will be an estimate of the covariance of the contaminating background (random noise or deterministic background such as twin-image). If local background gray level variations are significantly lower than gray level difference of an edge, then the filter performance will be good. This is the reason for low twin-image effect requirement, especially in the presence of multiple particles where the faster variations of twin-image of a particle can overlap with the edge of another one. Therefore, the twin-image suppression given in Chapter 3 is essential for good performance of the covariance filter.

If desired, the output image, $b(k,l)$, can be thresholded to get a binary image, which consists of a high level corresponding to edge pixels, and a low level for the rest.

4.4. Implementation and Results

The covariance filter given in previous section is implemented with $N=1$. The Fortran subroutine is given in Appendix D. Before display, the output array is quantized to achieve maximum dynamical gray range of 256, as in previous chapters. The following pictures show the results of edge detection by covariance filtering. The results are ready for various image analysis operations such as determination of particle shapes.

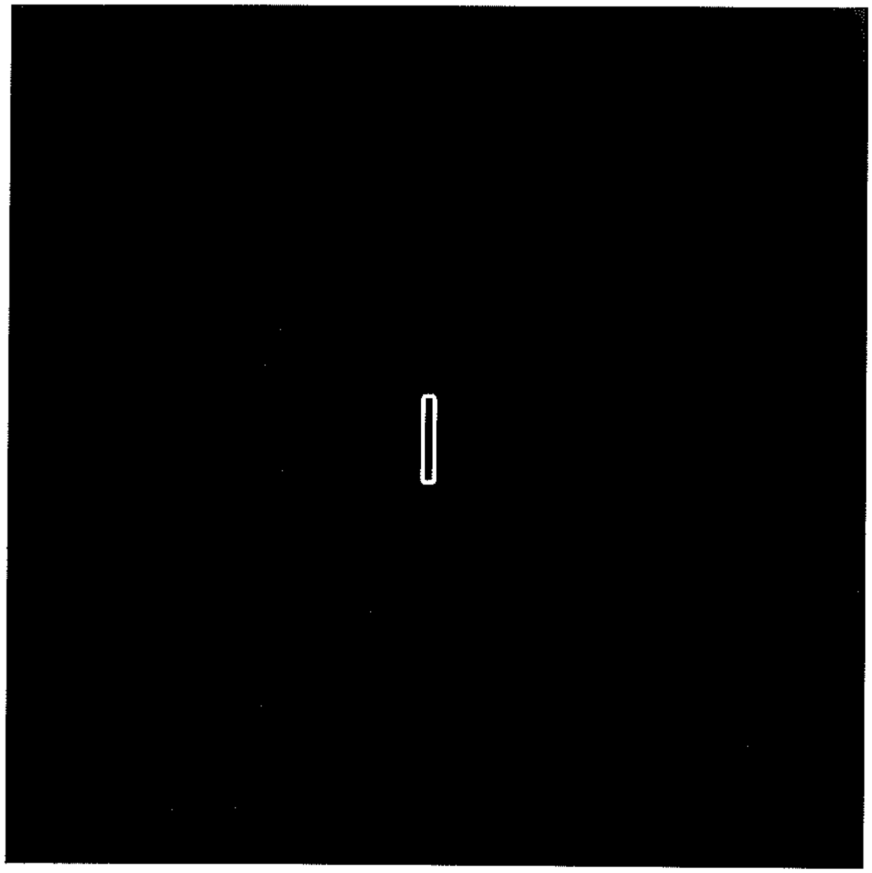

Figure 49. Result of covariance filtering applied to the conventional reconstruction of Fig.13.

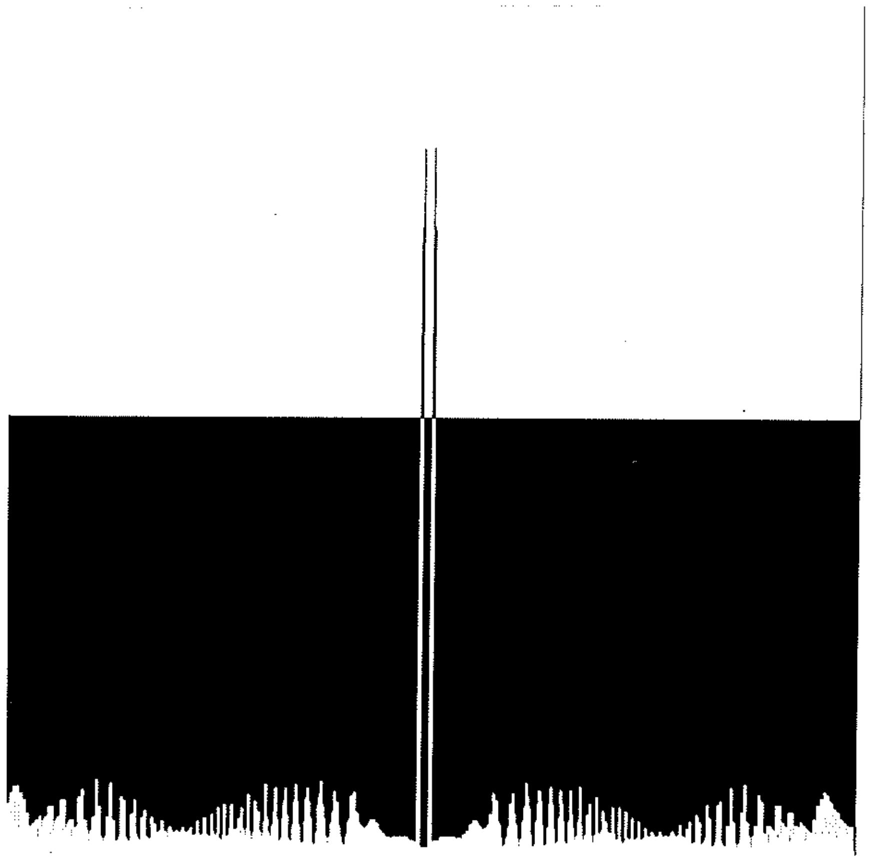

Figure 50. Profile of the image given in Fig.49 through its center.

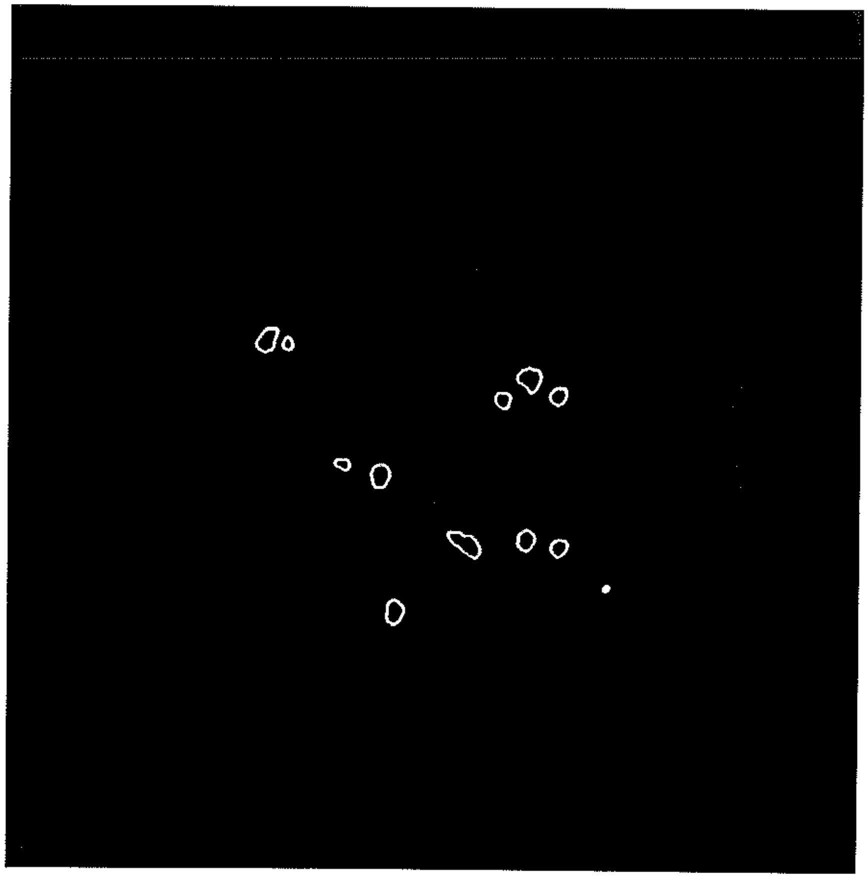

Figure 51. Result of covariance filtering applied to the reconstruction with twin-image elimination of Fig.43, after thresholding at gray level 128.

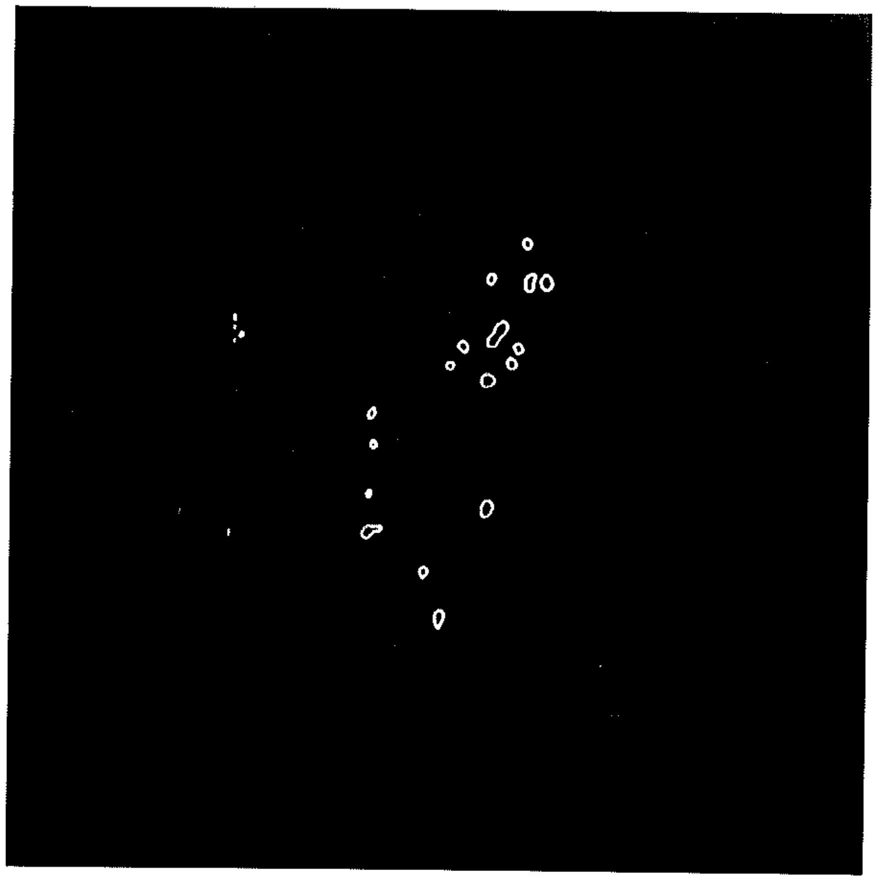

Figure 52. Result of covariance filtering applied to the reconstruction with twin-image elimination of Fig.44, after thresholding at gray level 128.