NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

C27185

AUTOMATIC MEASUREMENT OF
PARTICLES FROM HOLOGRAMS TAKEN
IN THE COMBUSTION CHAMBER
OF A ROCKET MOTOR

by

DENIS J.G. CARRIER

December 1988

Thesis Advisor: Prof. J.P. Powers

REPORT DOCUMENTATION PAGE		Form Approved OMB No 0704-0188

1a REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b RESTRICTIVE MARKINGS

2a SECURITY CLASSIFICATION AUTHORITY	3 DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited
2b DECLASSIFICATION/DOWNGRADING SCHEDULE	

4 PERFORMING ORGANIZATION REPORT NUMBER(S)	5 MONITORING ORGANIZATION REPORT NUMBER(S)

6a NAME OF PERFORMING ORGANIZATION Naval Postgraduate School	6b OFFICE SYMBOL (If applicable) 62	7a NAME OF MONITORING ORGANIZATION Naval Postgraduate School

6c. ADDRESS (City, State, and ZIP Code) Monterey, California 93943-5000	7b ADDRESS (City, State, and ZIP Code) Monterey, California 93943-5000

8a NAME OF FUNDING/SPONSORING ORGANIZATION	8b OFFICE SYMBOL (If applicable)	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER

8c. ADDRESS (City, State, and ZIP Code)	10 SOURCE OF FUNDING NUMBERS			
	PROGRAM ELEMENT NO	PROJECT NO	TASK NO	WORK UNIT ACCESSION NO

11 TITLE (Include Security Classification) AUTOMATIC MEASUREMENT OF PARTICLES FROM HOLOGRAMS TAKEN IN THE COMBUSTION CHAMBER OF A ROCKET MOTOR

12 PERSONAL AUTHOR(S) CARRIER, Denis J.G.

13a TYPE OF REPORT Master's Thesis	13b TIME COVERED FROM ____ TO ____	14 DATE OF REPORT (Year, Month, Day) 1988 December	15 PAGE COUNT 85

16 SUPPLEMENTARY NOTATION The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Hologram; Frequency histogram; statgraphics; Imageaction; Kolmogorov-Smirnov Test; Averaging; Mylar disk; Particle distribution

19 ABSTRACT (Continue on reverse if necessary and identify by block number)
This thesis describes the procedure used for the automatic measurement of particles from holograms taken in the combustion chamber of a rocket motor while firing. It describes the investigation done on two averaging techniques used to reduce speckle noise, capturing the image focused on a spinning mylar disk and software averaging of several image frames. The spinning disk and software averaging of several image frames. The spinning disk technique proved superior for this application. The Kolmogorov-Smirnov two-sample test is applied to different particle samples in order to find an estimate of the number of particles required to obtain a stable distribution function. The number of particles is calculated and given. The last part of this study shows real particle distributions in the form of frequency histograms.

20 DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21 ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a NAME OF RESPONSIBLE INDIVIDUAL POWERS, J.P.	22b TELEPHONE (Include Area Code) 408-646-2082	22c OFFICE SYMBOL 62

Automatic Measurement of Particles From
Holograms Taken in the Combustion Chamber of a Rocket Motor

by

Denis Joseph Gaston Carrier
Major, Canadian Armed Forces
B.Eng.Phy., Royal Military College of Canada, 1981

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December, 1988

ABSTRACT

This thesis describes the procedure used for the automatic measurement of particles from hologram taken in the combustion chamber of a rocket motor while firing. It describes the investigation done on two averaging techniques used to reduce speckle noise, capturing the image focused on a spinning mylar disk and software averaging of several image frames. The spinning disk technique proved superior for this application. The Kolmogorov - Smirnov two-sample test is applied to different particle samples in order to find an estimate of the number of particles required to obtain a stable distribution function. The number of particles is calculated and given. The last part of this study shows real particle distributions in the form of frequency histograms.

TABLE OF CONTENTS

LIST OF FIGURES

I.   INTRODUCTION

Performance of solid propellants in rocket motors is increased by the addition of aluminum and additives (aluminum oxide, zirconium, and so on).   Some of the original metal and metal oxides are found in the combustion chamber (from the surface of the propellant to the nozzle exit).   The increased propellant performance is related to its particle size distribution [Ref. 1].   In the past few years, research has been conducted at the Naval Postgraduate School in order to provide useful information about the particle size distribution.   This thesis is a continuation of that research.

In the approach studied, several steps are used to obtain automatic data retrieval from holograms taken in a small rocket motor combustion chamber. First, an optical hologram provides a 3-D image of the particles in the combustion chamber.   A real image of the image cross-section is focused on a transluscent screen (a piece of mylar) that can either be moving or stationary.   The hologram image is then recorded on tape, to be digitally processed later through different stages of filtering and analysis.   The hologram recording technique still needs to be improved if better resolution is to be achieved but, this will not be a topic discussed in this thesis; we will

only be concerned with the two later stages of processing the recorded images.

The image processing is done using an IBM PC/AT with both commercial image processing software and locally developed software. The first thesis on the image processing techniques were done by Redman [Ref. 2]. His work consisted in the capture of the image on a video recorder and a computer, and particle sizing of the holograms using the menu-driven ImageAction software from Imaging Technology Inc. He also developed Fortran programs to identify and size the particles. Computer memory size restricted him to work only with one quarter of the screen, but it was still taking him about four hours of computing time to process the image.

One of the main problem in the data retrieval is the creation of speckle noise during the laser reconstruction process. Different methods were tried in order to reduce the speckle. Redman used a spinning mylar disk during the image capture on the video tape. He also tried averaging several captured images using the ImageAction software. He found that both methods reduced the speckle noise by about the same amount but, since the averaging technique required more manipulation and more time than the mylar disk, the latter technique proved the better one to use, and was used in subsequent studies.

The speckle noise can further be reduced during the digital processing through different techniques. Edwards [Ref. 3] investigated the application of three nonlinear speckle reduction filters used in the Synthetic Aperture Radar (SAR) community: the geometric filter, the sigma filter, and the local-statistics filter. He concluded that the geometric filter had the best overall performance, but that in some cases the local statistics filter could perform better. He suggested that in processing an image both methods be tried and that the best result be used. Edwards's thesis also introduced the use of the ITEX/PC software, also from Imaging Technology Inc. This software permits us to write Fortran, Pascal, and C language programs that call up subroutines which perform most of the ImageAction functions and more. Filtering to remove speckle usually causes the resolution to degrade. Without filtering, resolution of about 12 micrometers could be obtained from the image recorded with the spinning mylar disk method but, after filtering, the best resolution that Edwards could reach with any of the three filters was 14 micrometers.

Orguc [Ref. 4] used two locally developed filters, the 5 x 5 Convolution filter, and the 3 x 3 Gaussian filter, in order to increase processing speed and have better speckle reduction. His locally developed filters turned out to be much faster than the previous ones, but were poorer in

3

speckle noise reduction. Because of this, he recommended that the geometric filter be used during further studies.

Once the filtering is done, the image is thresholded in order to separate the particles from the background. The final steps are the particle identification, counting, and sizing.

Orguc greatly reduced the feature data extraction time by improving the Fortran routines that were used, and by using for the first time a fast access memory board and a 16 MHz 386 microprocessor. A full screen image could be processed in about ten minutes, compared with close to 4 hours for a quarter of a screen for Redman. Orguc also introduced the use of the software package, Statistical Graphics System (STATGRAPHICS), by Statistical Graphics Corporation. He used that program to produce the histogram of the particles size distribution from the data output from his Fortran routines. (In this study STATGRAPHICS is also used to create frequency histograms of the particles horizontal coordinate in order to perform the Kolmogorov-Smirnov statistical test.) Orguc suggested in his study that approximately 1000 particles are a sufficient number to reach a representative particle size distribution. Fewer particles provided an incomplete distribution. More particles produced only minor changes in the size distribution.

This thesis has the following objectives: to find out if the speckle index can be further reduced by averaging a certain number of frames, to find out how many particles are required in order to have a stable particle distribution, and finally to obtain a typical particle distribution in the combustion chamber of a small rocket motor. This report is divided into six chapters. Chapter II introduces the steps taken in the hologram image processing, the image acquisition and enhancement, and the data manipulation, which includes the particle identification and sizing and counting. Chapter III introduces two speckle reduction techniques, the spinning mylar and the software averaging of images. Their performance and results are compared and analyzed. Chapter IV uses the Kolmogorov-Smirnov two samples test in order to compute the number of particles required for a stable distribution within a user specified confidence level. Chapter V gives a typical particle distribution of a small rocket motor while firing. Chapter VI contains the closing remarks.

II. HOLOGRAM IMAGE PROCESSING

Different techniques are used to obtain a 3-D image of the particles in a rocket motor. In this study, as mentioned in the introduction, we use a holographic technique which gives us the opportunity to record the particle distribution of the entire depth of the field. The hologram is recorded using a Q-switched pulsed ruby laser and reconstructed with a krypton laser. The recording process will not be further discussed here but a good description can be found in Reference 1.

In the reconstruction process the image is recorded on a video tape to be later processed for digitization, enhancement, and data manipulation. A more detailed explanation of each processing step is given in the next few sections. The hardware and software used for the entire process from the image acquisition to the histogram of the particle distribution may be summarized as follows:

1. Hologram reconstruction subsystem (hologram, krypton laser, mylar disk).

2. Image recording system (microscope, video camera, video tape recorder).

3. Image processing hardware system (IBM-PC/AT, 16 MHz 386 microprocessor board, PC Vision frame grabber board, video monitor)

4. Image processing software (ImageAction, Itex/PC, STATGRAPHICS, locally developed programs).

In this thesis, the holograms of two test images were used and the remainder of the holograms were of real particles in the combustion chamber of the rocket motor while firing. One of the test images is a 1951 USAF standard resolution chart reproduced in Figure 2.1. It is used in Chapter III where the software averaging of numerous images are compared with the mylar disk averaging effects. It was used primarily because of its features of known sizes which would permit us to evaluate more easily the effects of each iteration on resolution.
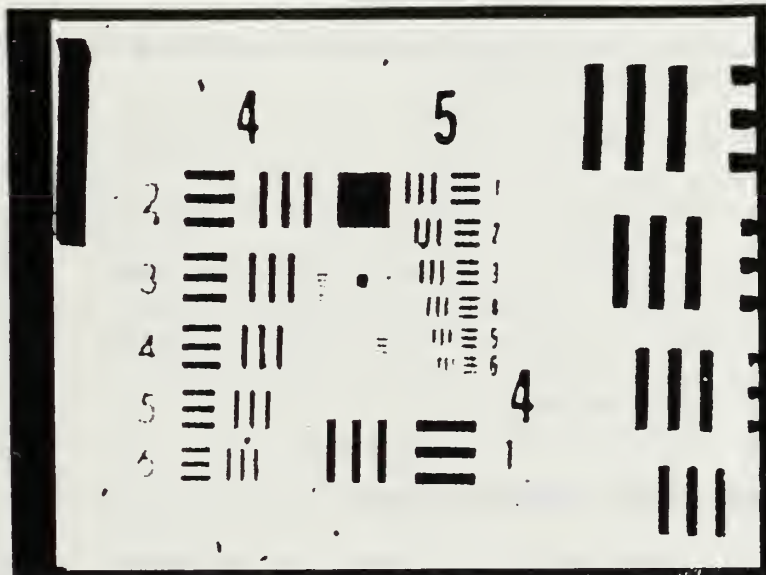
Figure 2.1 USAF Standard Resolution Chart (without speckle noise).

The following is a resume of the steps required in order to obtain the histogram of the particle distribution from the constructed hologram:

1. Hologram reconstruction and image reproduction (including speckle reduction using mylar disk).

2. Image storage on videocassette.

3. Image digitization and storage on the computer system (using ImageAction software).

4. Speckle noise reduction through averaging (if effective).

5. Speckle noise reduction through filtering.

6. Separation of particles from background through thresholding.

7. Particles identification (using locally developed programs and Itex/PC software).

8. Particles sizing (also using locally developed programs and Itex/PC software).

9. Production of particle distribution histogram (using STATGRAPHICS).

All of the above steps, with the exception of step 4, are used in sequence to obtain the histograms. Each step uses the final product of the previous step, so they have to be performed in sequence. We now consider each step of the process in more detail.

A. HOLOGRAM IMAGE REPRODUCTION

A pulse ruby laser is used to construct the hologram which is placed on a transparent glass plate. A krypton laser, which operates at a wavelength very similar to the one of the pulsed ruby laser (.6471 and .6943 micrometers

8

respectively), is used to reconstruct the 3-D image. Figure 2.2 illustrate the set-up used to record various reconstructed images on a video cassette recorder. As we know from holography theory, illuminating a hologram film with a laser light will produce a 3-D image of the original physical situation.

Using a microscope with magnification power of one, two, four or ten, the image can be magnified and then be recorded on a video cassette recorder (VCR) through a video camera recorder. The VCR is required due to the remote location of the hologram reconstruction equipment from the image processing computer. Higher magnification levels produce smaller depth of fields, so focussing becomes harder. A greater magnification also produce a smaller field of view, so less light is available to the camera. This in turn requires more sensitivity in the camera or higher power from the laser. It would seem that a smaller magnification level is desirable, but another factor comes into play, the size quantization error. The particle size is given by its number of pixels. The larger the particle image, the smaller the quantization error will be [Ref. 4]. For a smaller quantization error we would require a higher magnification level. A higher magnification level means fewer particles per image thus requiring more images for the same number of particles, meaning a greater processing time. We have to compromise between these system features;

9

the magnification power of four was found to be the best level to work with [Ref. 4]. A magnification power of four was consequently used throughout this study.

One of the main problems we faced was the speckle noise. It is created in the recording process because of the requirement to remove phase stiration due to thermal gradients within the motor. One method used to reduce this speckle is to place a spinning mylar disk diffuser at the location of the reconstructed real image as shown in Figure 2.2. Chapter III discusses the use of the mylar disk and a possible replacement technique, the averaging of several image frames, in detail.
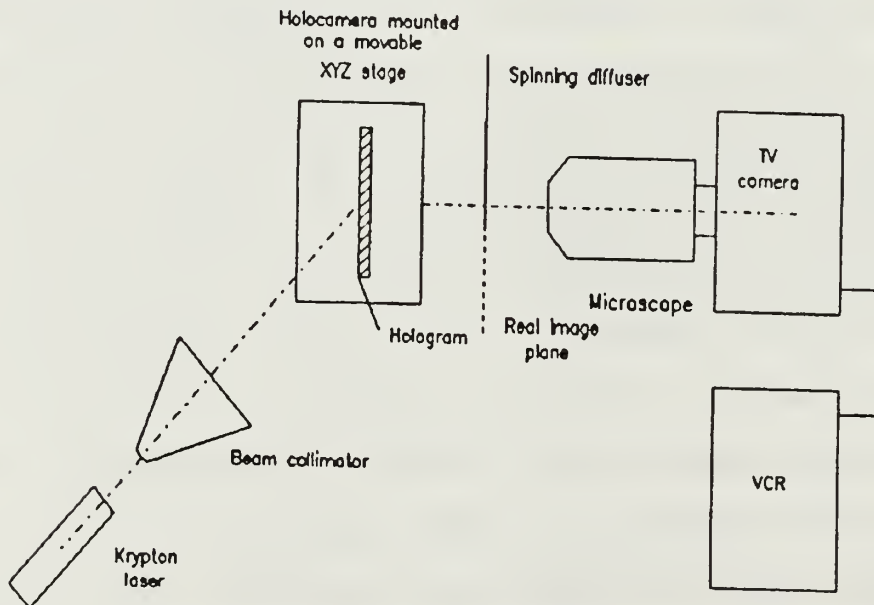
Figure 2.2 Hologram reconstruction setup.

Now the image is saved on tape and ready to be stored on the computer and to be enhanced using different techniques which are discussed in the next section.

B. IMAGE ACQUISITION AND ENHANCEMENT

The image is ready to be digitized and stored on the computer (either on a hard disk or a floppy disk or both). The IBM PC/AT with a 16 MHz 386 microprocessor board and a PC Vision frame-grabber board are used to execute these operations. The PC Vision board assigns a grey value between zero and 255 (black to white) to each pixel of the image. The image which is displayed on the television screen is made up of 512 X 480 pixels (the last 32 rows on the screen cannot be seen). An image of the USAF standard resolution chart (before any enhancement and without the mylar disk) is shown in Figure 2.3 along with a histogram of the image. The horizontal axis of the histogram gives us the grey level which varies from 0 (black) on the extreme left to 255 (white) on the extreme right. The vertical axis gives the total number of pixels which has the specified grey value. In the next sub-sections we will see the effect of several improvement techniques on the image and its histogram.

1. Software products and image acquisition

Two commercial software packages are used in this study along with several locally developed programs which

11

we will discuss in more detail later. The ImageAction software package is under control of its operator through a series of menus. The user chooses which operation to perform using a mouse. The package provides image processing, image graphics and image analysis capabilities. Some of the most useful functions are the GRAB function which permits the image to be digitized as discussed earlier, the SAVE SCREEN function which stores the digital image to a hard disk or a floppy disk as desired, and the RESTORE SCREEN function which permits the user to bring the image that had been saved on disk back on the screen (and frame grabber board). A more detailed discussion of the ImageAction software by Imaging Technology Inc. can be found in reference 5.

The second commercial software package which is used extensively in the study is the Itex/PC package also by Imaging Technology Inc. [Ref. 6]. This program can perform most of the functions that ImageAction can, and more. The main difference between these two programs is that Itex/PC is made up of subroutines that can be called from a Fortran, Pascal or C program. The subroutines are used extensively in all locally developed programs. Those subroutines can be used to calculate and return a value or can be used to perform a function. For example:

    errval = readft (ixs, iys, ncol, nrow, fname, comline)
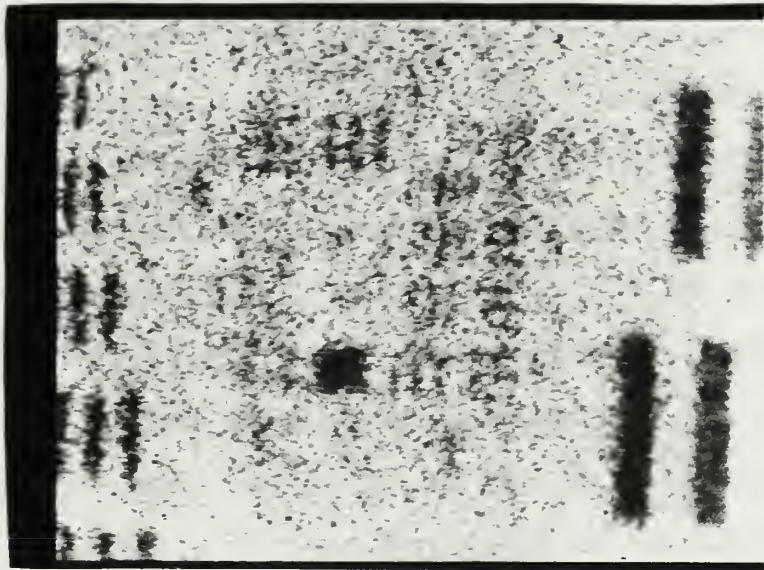
(a)
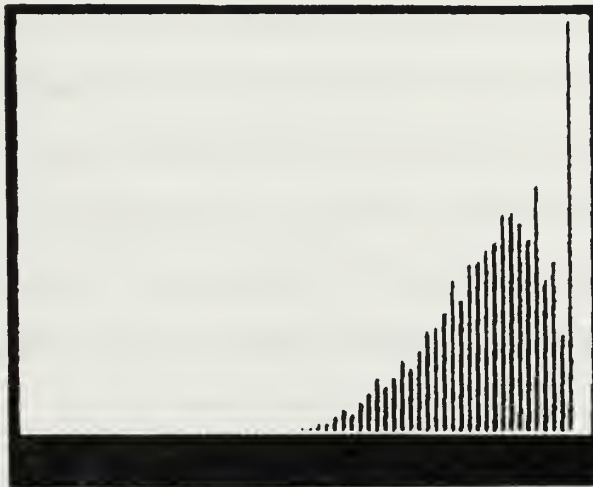
(b)

Figure 2.3   USAF Resolution Chart with speckle
noise, no enhancement, (a) image,
(b) histogram.

13

reads the file called fname from disk, and writes it on the frame grabber board and the television screen. Ixs and iys are the horizontal and vertical coordinate respectively of the upper left-hand corner of the area on the frame-grabber board where we want the retrieved image to be placed. Ixs varies from 0 to 511, and iys varies from 0 to 479. Ncol and nrow indicate the horizontal and vertical size respectively of the area in which we want the image placed. Ncol can take the values from 0 to 511, where nrow can take the values between 0 and 480. Comline is the name of the variable to receive the comment line associated with the image. Errval is the error return code.

With the image on the computer, the next step is to try to improve the image to permit the minimum amount of error in the sizing and counting processes. We have already mentioned that our biggest source of error is caused by the speckle noise. Consequently our strongest efforts are spent trying to reduce this speckle noise. The next two subsections discuss possible techniques for these improvements.

2. Image averaging and mylar disk

The speckle noise can be reduced by averaging images with differing speckle patterns. The averaging can be done various ways, several frames of an image can be averaged using the ImageAction program or using a locally developed program. Also a spinning mylar disk can be

inserted in the reconstruction setup. These techniques
will not be discussed further here since they are discussed
in details in Chapter III, where performance of the
spinning mylar disk and the averaging of several image
frames are compared and analyzed.

3. Filtering

There has been an enormous amount of effort spend
in previous studies to reduce the speckle noise through
filtering. Speckle noise is not a new problem; the SAR
community has developed several techniques to reduce the
speckle noise. The ImageAction package has two routines
that can be used for speckle reduction (the LOWPASS and FEW
routines). Edward [Ref. 3] has implemented three filters
which are used in the radar community: the geometric
filter, the sigma filter and the local statistical filter.
He concluded that the geometric filter in general gave
better results and should be used for our purposes. Orguc
[Ref. 4] for his part implemented two locally developed
filters, the 5 X 5 convolution filter, and the 3 X 3
gaussian filter. They were found to be faster than the
previous three, but since their resolution degradation was
high and speckle reduction not as good, the geometric
filter was again found to be the best one. The
disadvantages of using these filters is that they all tend
to degrade the resolution, hence blurring the image. They
also all require some time to run which slows the process

down.   In this study the speckle reduction capabilities of
the geometric filter is studied when the filter is used
with and without the spinning mylar disk, and with a
software averaged image.   A detailed explanation of these
filters may be found in References 3, 4, 7, 8, and 9.

4.   Thresholding

As we can see in Figure 2.3b, the histogram of an
image at that point does not differentiate too well between
a particle and a speckle.   This is where the thresholding
of an image becomes necessary. The thresholding is a
process that will separate the particle from the background
(noise).   Figure 2.4a shows the histogram of an ideal case,
where the object (or particle) has very different grey
levels than the background.   In this case, setting the
threshold to a grey value in between the particles and the
background values will separate the two clearly with no
ambiguity.   As it is always the case the real world is not
all black and white, but different shades of grey as shown
in Figure 2.4b.   In most cases the grey level of some
particles will be greater than the grey level of some
background noise.   So here, setting a threshold will
introduce some error because some pixels of the background
will be counted as objects and some object pixels are
omitted.   As we can see, thresholding will introduce a
certain amount of error; the better the threshold value the
smaller the error will be.
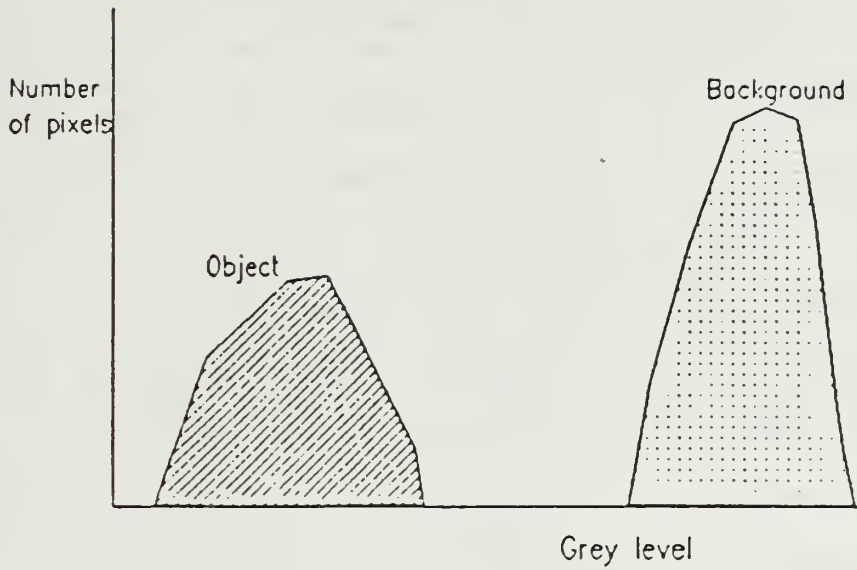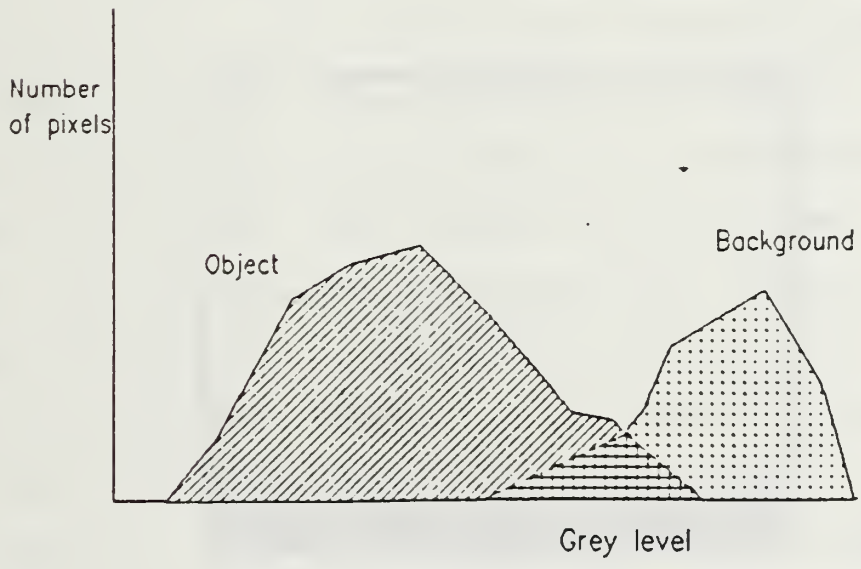
Figure 2.4   Histogram of an image, (a) ideal case,
(b) real case.

The threshold operation is done using a program written by Edwards [Ref. 3]. He wrote a subroutine that can be called from a Fortran program, and uses the Itex/PC subroutines. It was later modified so it could be used as a program. The operator sets a threshold value, and all pixel values greater than the threshold value are set to 255 (white) and all pixel values smaller than the threshold, are set to 0 (black). In this way all that is left in the image are dark features (black particles) and white (background). Figure 2.5 shows the histogram of a thresholded image, as we can see the only thing left are the particles (0 value) and the background. To visualize

Figure 2.5 Histogram of thresholded image.

the effect of some enhancement techniques we have included two images with and without the enhancement techniques, shown in Figure 2.6. As you can see there is an enormous improvement in the image when the spinning mylar disk, the geometric filter and the threshold techniques are applied. The image is now in proper shape so the features could be sized and counted in order to produce a histogram of their distribution.

5. Programming in C language

Before we go to the data manipulation section we should write about the speed of the process.

There are two properties that are very important in this speckle reduction, the resolution and the processing speed. Both aspects were improved a great deal since Redman's work. The resolution was improved through better hologram and speckle reduction in particular. The greatest improvement was achieved in the speed of the process, through hardware improvement (the addition of the 16 MHz 386 processor in particular) and through software improvement (Itex/PC package). The speed was increased from about four hours to processing one quarter of screen with Redman to about ten minutes for a full screen now. The next logical step was to write all the Fortran programs in the C language as is currently being done by another student [Ref. 10].
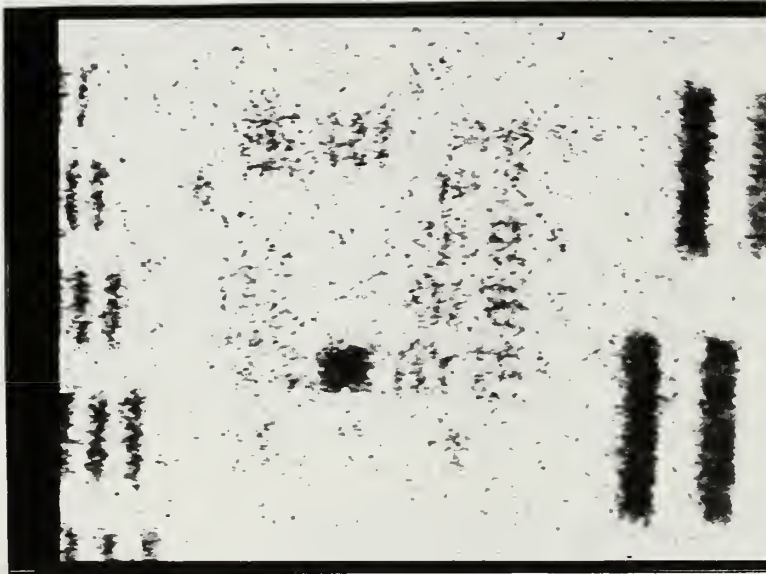
(a)

(b)

Figure 2.6   Effect of enhancement techniques on a processed
image, (a) image with no speckle reduction and
no threshold applied, (b) Image with spinning
mylar, geometric filter and a threshold used.

C. DATA MANIPULATION

The goal of this research is to computerize the process of sizing and counting the particles distributed in a rocket motor's combustion chamber with minimum error. Two programs were developed by Redman and improved on by Orguc, to accomplish those two latest processes. The results are displayed in the form of a histogram of the particle size distribution using the commercial software package STATGRAPHICS. The next three subsections explains in more details those three steps of identification, sizing and counting, and results display.

1. Particle identification

We now have the thresholded image with pixel values of either zero or 255, and are ready to identify the particles on the image. The feature identification program is an interactive program which first asks for the name of the image file to be loaded. Once the file is loaded, the program sweeps the screen and reads each pixel in such a way that adjacent pixels are labeled with the same feature, and each feature is assigned a feature number from one to a maximum of 255. If there are more than 255 particles, the particles above 255 are assigned a number between, one and 255 again but with a different group number to differentiate them. The program requests an estimate of the maximum feature size on the screen from the user. The user gives an approximate value in terms of pixels. A good

value for an image of the combustion chamber is around 70. The larger the value given, the longer the program will take to run. The program terminates by giving the number of particles found.

2. Particle sizing and counting

This program uses the results of the previous one, and is used to find the specific dimensions of each labeled feature. It calculates the horizontal and vertical diameters of each particle along with its total area.

Also an interactive program, it starts by asking the user for the number of features there are on the screen. This value is supplied by the previous program. The program also requires the magnification level of the microscope. This is needed since the program calculates the feature size in micrometers and a different magnification level means a different scale to operate with. The last thing the program will request is the maximum feature length, which obviously is the same value supplied to the first program.

The output of the program is a file in the form of a table. The table has four columns which gives the feature number, the feature total area, its horizontal diameter, and its vertical diameter in micrometers. Appendix A is an example of a typical output file. For more details on the operation of these programs see References 2 and 4.

22

3. Results display and analysis

We now have the desired features and their sizes, and we want to display them in a manner that is easy to see and analyze as the final step of our analysis. The display format was chosen to be a frequency histogram and the program used to do it is STATGRAPHICS. This package was chosen by Orguc for its compatibility with the previous output file, and the IBM PC/AT system. It was also retained in this thesis for its statistical capacity, in particular its capacity to produce frequency histograms, which is used in Chapter IV.

STATGRAPHICS is a statistical and graphics package which is very powerful; only a small fraction of the program is used in this thesis. We now look at the main features of the package. The data management utilities are used to import data files, display, export, and to execute file operations such as adding data files cumulatively, which permits us to display the combined statistics of several images. The descriptive methods section permits us to create and display the histogram of the particle distributions. It also reproduces frequency histograms which are used to apply the Kolmogorov-Smirnov two samples test described later.

III. SPECKLE REDUCTION BY AVERAGING

We have seen that there are many different ways of reducing the speckle noise. Edwards and Orguc used different filtering algorithms. In Reference 2, Redman introduced the use of a spinning mylar disk. He also tried an averaging technique using the ImageAction package. He found that the spinning mylar disk had much better performance. Following that analysis, other users all used the spinning mylar disk as a first stage of speckle noise reduction. It was felt that speckle reduction through filtering had reached close to its maximum efficiency [Refs. 3 and 4]. One of the main objectives of this study was to compare the classical averaging techniques as a baseline for comparison of the other nonlinear filters. We tried to replace the spinning mylar disk by an averaging technique, using a locally developed program. As we will see later, the mylar disk turned out to be the best performer in most ways.

A.   SPECKLE AND SPECKLE INDEX

Speckle and speckle index are two very important notions in holography. They are specially important for this thesis because of our objective to try to reduce the speckle noise by an averaging technique. References 1, 3,

24

4, 7, 8 and 9, described speckle and speckle index, but due to the importance of those two notions they will be reviewed in the next two subsections.

1. Description of speckle

During the laser reconstruction of the image, speckle noise is introduced in the image. The speckle size can be as big or bigger than the smallest particle size, and can introduce a non-negligeable error in the particle identification and counting process. This is why the speckle noise is one of the main problems in this area of study. The effect of speckle noise can easily be seen in Figure 3.1. Some particles appear in the image with speckle noise where we know there are no particles, since those "noise particles" can be of the same or greater size as the real particles.

This speckle has the characteristic of random multiplicative noise [Ref. 9]. Models of this type of noise exist but are not very precise. Well known to the SAR community, this group proposed two different methods to reduce the speckle noise. The first method is by image filtering, which was done and optimised by Edwards and Orguc. The second method is by averaging successive frames of the same image, which is the subject of this chapter.

2. Description of speckle index

The speckle index is a figure of merit to determine the amount of speckle in an image. Crimmins

[Ref. 8] showed that a reasonable measure for the speckle index could be the ratio of local deviation to local mean. He proposed the following algorithm.

The local deviation is defined as

$$\sigma(m,n) = \max \{F(m+a,n+b)\} - \min \{F(m+a,n+b)\} \qquad (2.1)$$

$$-1 \leq a,b \leq 1$$

and the local mean is

$$\mu(m,n) = \frac{1}{9} \sum_{a,b=-1}^{1} F(m+a,n+b) \qquad (2.2)$$

The speckle index is then given by

$$\text{speckle index} = \frac{1}{MN} \sum_{m=1}^{M} \sum_{n=1}^{N} \frac{\sigma(m,n)}{\mu(m,n)} \qquad (2.3)$$

In the above equations, M and N are the dimensions of the local area in which the speckle index is calculated, F(m,n) is the gray level of an individual pixel located at m,n.

Edwards adopted this algorithm and wrote a Fortran subroutine to implement it. He choose M and N to be equal to 240 as a good representation of the image and yet does not take on excessive amount of time to calculate. The same program was used in this thesis for speckle index calculations.

26
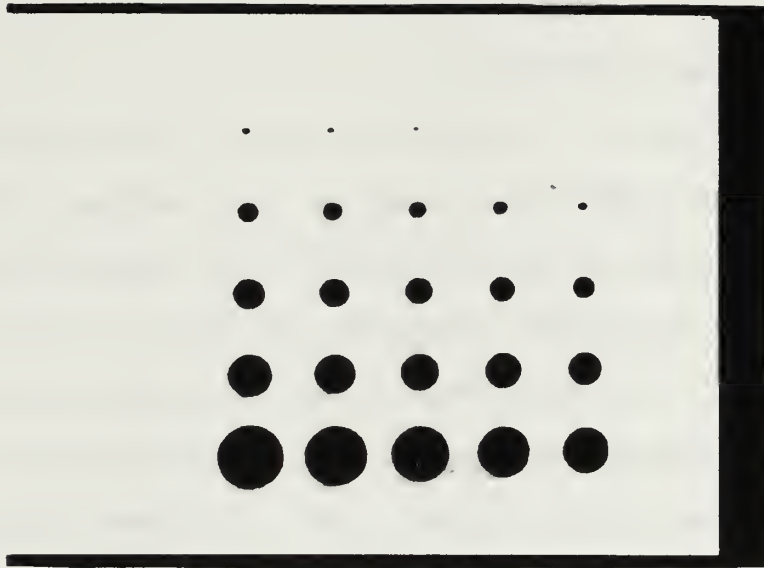
(a)

(b)

Figure 3.1 Sample image, (a) without speckle,
(b) with speckle.

B. AVERAGING TECHNIQUES

Redman [Ref. 2] proved that the use of the spinning mylar was contributing greatly to reducing the speckle noise. He also used the ImageAction averaging routine to take successive image averages in order to average their grey levels. He concluded that the spinning mylar was a better solution method to use due to its readily available results for about the same amount of speckle reduction as the averaging routine.

Redman's measurements of the speckle reduction was all done by visual qualification, since he did not have any means to calculate the speckle index. Now with the tools we have available, we decided to quantify his results. But after careful reading of the ImageAction averaging routine, it was found that it was not doing exactly the averaging of image frames as we wanted. The averaging routine in ImageAction averages several images over time. It acquires a series of sequential images supplied by the operator, sums the values of each pixel in the AOI, and divides by a number supplied by the operator. The problem with this function is that the sequential images are acquired very quickly, faster than we can supply images of different speckle pattern. Because of this the averaging routine in reality averages the same speckle pattern over different images. Two Fortran routines were written in order to average images of different speckle pattern. The next two

28

subsections discusses the two techniques and how they were applied.

1. Mylar disk

The mylar disk is used during the hologram reconstruction phase. It is placed in the path of the real image in front of the microscope as shown in the hologram reconstruction setup of Figure 2.2.

The rotating disk changes the phase of the image over time, and this will change the speckle pattern seen by the camera. If the disk rotates faster than the one-thirteeth of a second integration time of the video camera, the camera tube will perform the averaging process. Since the particle positions are fixed, the speckle values tend to average to a middle pixel value (grey) whereas the particles will tend to stay darker (small value). In this way the speckle noise is filtered out. Visual inspection of the image and speckle index calculation demonstrates clearly the effectiveness of the rotating mylar disk on speckle noise reduction. The relative speckle index was reduced from a value of 1.0 to about 0.20 on the image of the USAF Resolution Chart. The main advantage of the technique is that it requires no processing time to be implemented.

2. Software image averaging

The image frame averaging done by software is naturally accomplished once the images have been digitized

and stored on the computer. Two Fortran programs that use the Itex/PC subroutines extensively were written for this part of the study. Two programs were required because of the limited amount of random access memory (RAM) on the computer system. The full screen could not be averaged because of this memory limitation, so only one half is done at the time. Frames are the images which are to be averaged. Each frame is an image of the same particle distribution in the combustion chamber, but with a different speckle pattern. The different speckle patterns are obtained by randomly rotating the mylar disk for each frame. By averaging the different frames we smooth out the speckle noise while the particles stay the same. The first program divides the images to be averaged in half and stores them separately. The second program loads and averages one half of all the frames, then does the same thing for the other halves and finally saves the fully averaged image to disk.

The first program may be found in Appendix B. Since the program is fully automatic with the exception of two questions asked to the user, the image files must already be stored with the names, IMG01.IMG, IMG02.IMG, and so on. The two questions asked are for the number of images the user wants to average (or separate in two halves) and whether the user wants to compress the images to be stored. The program sets the area of interest (AOI)

to full screen (512 columns by 480 rows) and then reads the first image using the Itex/PC subroutine IMREAD. The next steps divide the AOI into two half screens and save each half in different files such that the left half of the image is stored as LHS01.IMG and the right half as RHS01.IMG. The program repeats this process for the number of frames to be divided and terminates after stating the number of images that have been divided and saved.

The second program, which may be found in Appendix C is a fully automatic program which uses the files created in the previous program. It requires only the number of images you desire to average. An array of 256 columns by 480 rows is created. Each pixel value is then read and added to its proper position in the array. The reading portion is done using the Itex/PC subroutine RPIXEL. Once this step is completed for all images, each pixel value is divided by the number of images. In this way the pixels which represent a part of a particle (which should have the same value for each image) should average to that value. On the contrary, the pixel of noise which should vary from close to white (225, for example) to a darker grey (like 150 or 200) should average to a lighter value than the particles. This way the gap which differentiates between particles and noise should become wider. Once this is done, the same processing is repeated for the other halves. The two averaged halves are then joined together and stored

to disk with the file name AVGIXX.IMG, where XX is the number of image frames averaged. It takes about 25 minutes to average 15 frames, including dividing the images into two halves.

The effect of this averaging routine on the test image and its histogram can be seen in Figure 3.2. The histogram of the non-averaged image shows us a gradual distribution of the pixel values, showing no discernable difference between a particle and the background or noise. It still is not easy to chose a good threshold value from this histogram to separate what is a particle and what is noise. The histogram of 25 averaged image frames shows us a cleaner difference between noise and particles. The averaging routine brought down the higher pixel values of the noise and increased the value of lower pixel values, which is not desirable. But it also separated the particles from the background by grouping the particles together and the speckle noise together. The particles can be recognized by the small bump followed by the plateau on the left of the histogram. The choice of a good threshold value which should be set in the middle of the plateau, would permit us to separate particles from noise. The averaging technique thus does what we expected and could be used further on.
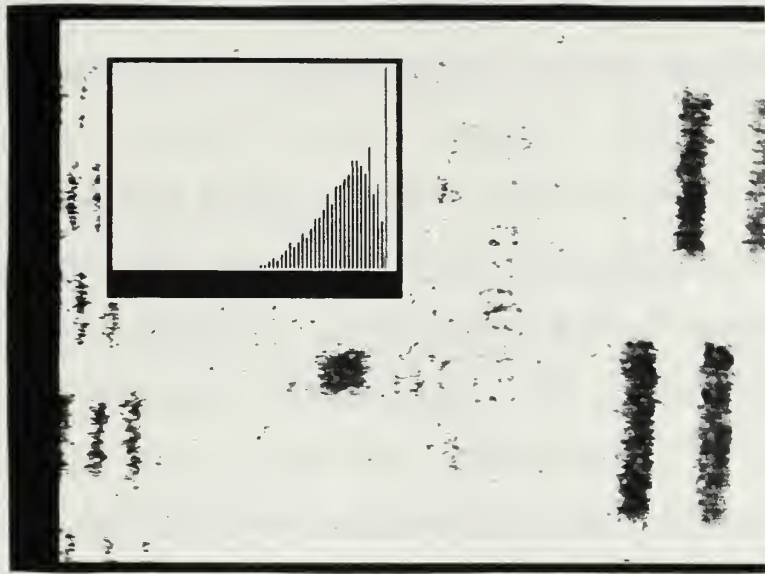
(a)
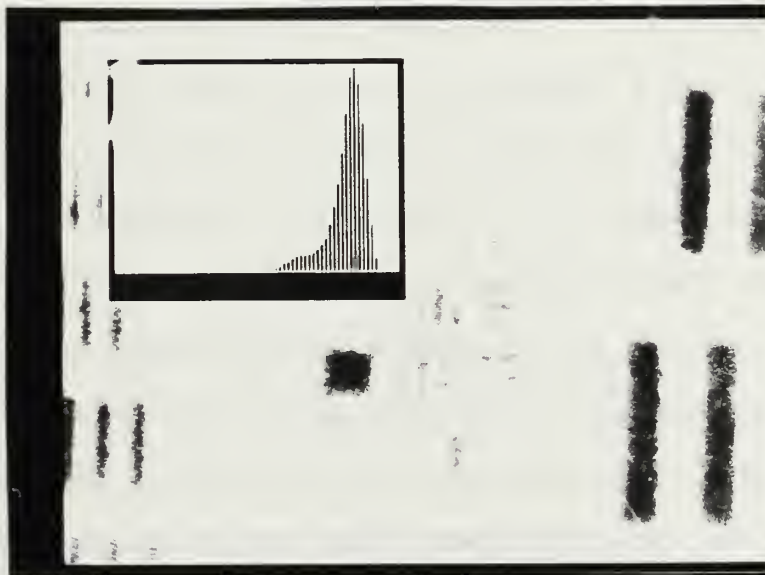
(b)

Figure 3.2 Image and histogram of USAF chart, (a) no
averaging, (b) 25 images frames averaged.

C.   COMPARISON OF THE TWO AVERAGING TECHNIQUES

We now will compare the ability of the averaging techniques to reduce the speckle index and their effect on resolution. Finally we will consider the processing time and will look at the possibility of using a combination of both techniques.   We will compare the results for those obtained from the geometric filter.

1.   Speckle index reduction capabilities

The USAF resolution chart was used for these measurements.   All speckle index calculations were done directly after averaging; none of the filters (geometric, sigma or local statistics) were used.   Figure 3.3 shows the results obtained.   The plot gives the relative speckle index value as a function of the number of averaged image frames for both the mylar and the averaging techniques.

The theory of averaging random multiplicative noise, like the speckle noise in our case, says that its reduction should be inversely proportional to the square root of the number of images, which is represented by the broken line.   The measured values follow the theory very closely.   We can also see that after averaging about 15 frames, the speckle index stabilizes at a relative value of about 0.25 (which is still greater than the 0.20 value for the spinning disk that is also shown in the figure).

We concluded from these measurements that our averaging routine is effective in reducing the speckle

34

index as expected from the theory, but that the spinning mylar disk is still a more effective way to reduce the speckle index.

2. Resolution degradation

We already mentioned that averaging the image had the effect of increasing resolution. As the larger speckle is reduced, smaller features become recognizable. Measuring the resolution of an image is a subjective measurement. We used the USAF Resolution Chart, which has
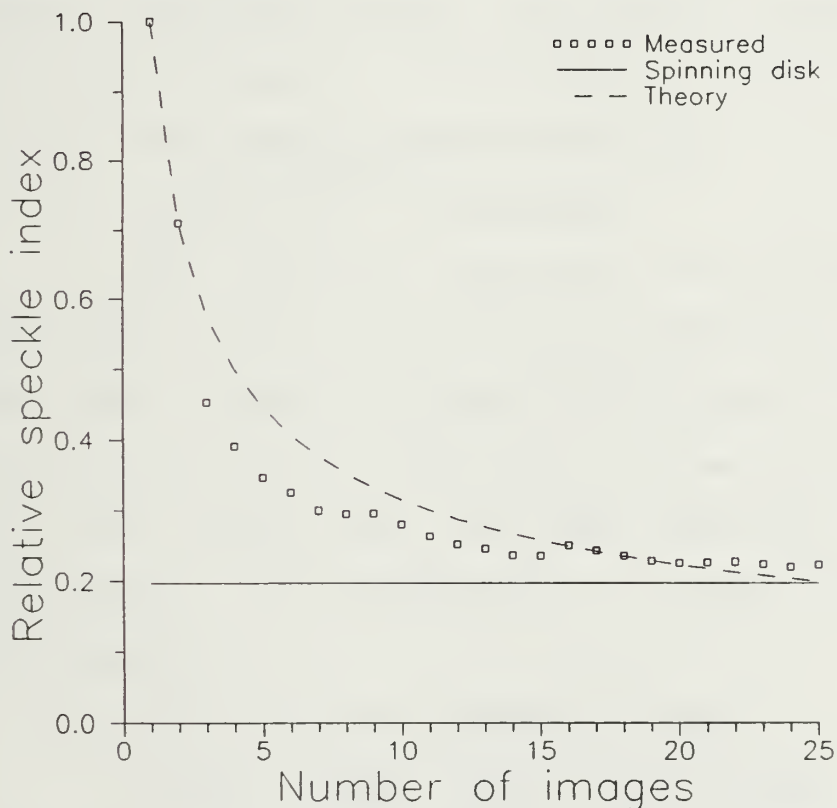
Figure 3.3   Speckle reduction by image averaging.

known bar sizes, and judged the smallest bar that we could recognize for each image.

The results of these measurements are found in Figure 3.4. As we can see, the averaging routine does improve the resolution, then stabilize after averaging 15 frames to a value of about 17 micrometers. But here again the rotating disk performs better. Without any further processing, a resolution of about 12 micrometers is obtained with the mylar disk.
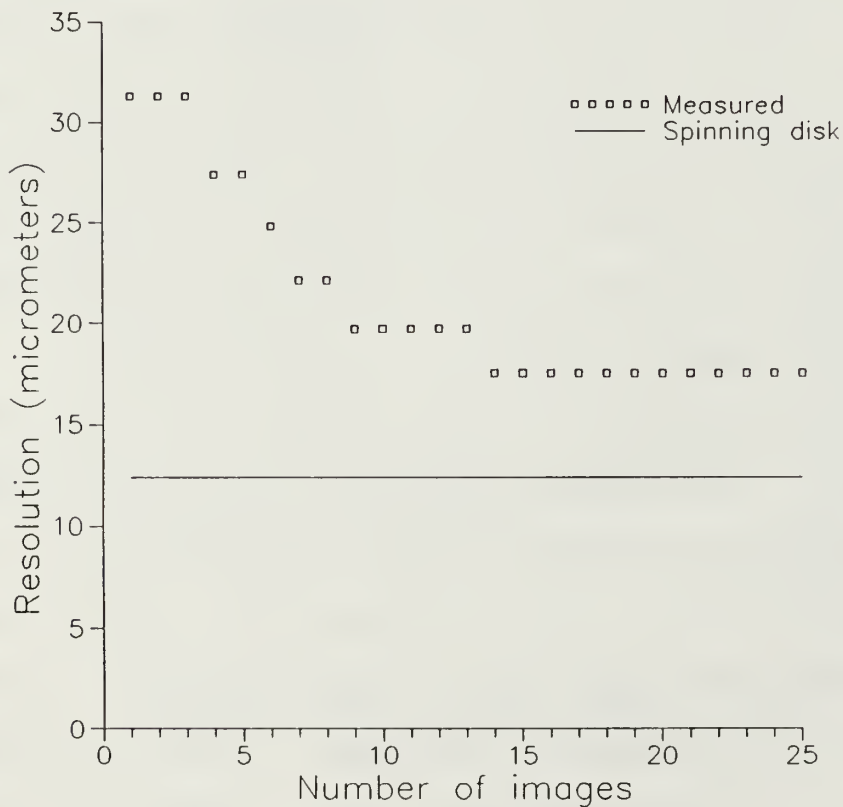
Figure 3.4 Resolution after image averaging.

36

3. Time factor

As we mentioned in Chapter II, trying to increase the processing speed is important. This is another area where the spinning mylar has a big advantage because the results from it are available in the averaging time of the recording camera (1/30 s) compared to several minutes to get the results using the averaging technique. The averaging technique also requires more interaction with the user because he has to save each individual frame using ImageAction, then divide them into two halves using our locally developed program, and finally average them with the second program. Not only does it take computer time, it also takes significant operator time.

4. Combination of techniques

The averaging technique was applied to images which had already been averaged by the mylar disk. It was found that further averaging had negligible effect on the speckle index and the resolution of the image. Considering the little effect and the added time and manipulation requirements, this combination of both methods is not recommended.

5. Effects of the geometric filter

Before making conclusions on the use of one technique or the other, we wanted to know what kind of performance the geometric filter had on each of the images resulting from the use of the methods. We ran up to nine

iterations of the geometric filter on the image averaged with the spinning mylar disk, and on a 25-frame averaged image. We also included the effect of the geometric filter on a non-averaged image. The results are shown in Figure 3.5. We can see that the best speckle noise reduction is obtained when the geometric filter is applied to an image of 25 averaged frames. A very close second is the image averaged with a mylar disk and last, as expected, is the image without any averaging technique applied to it.
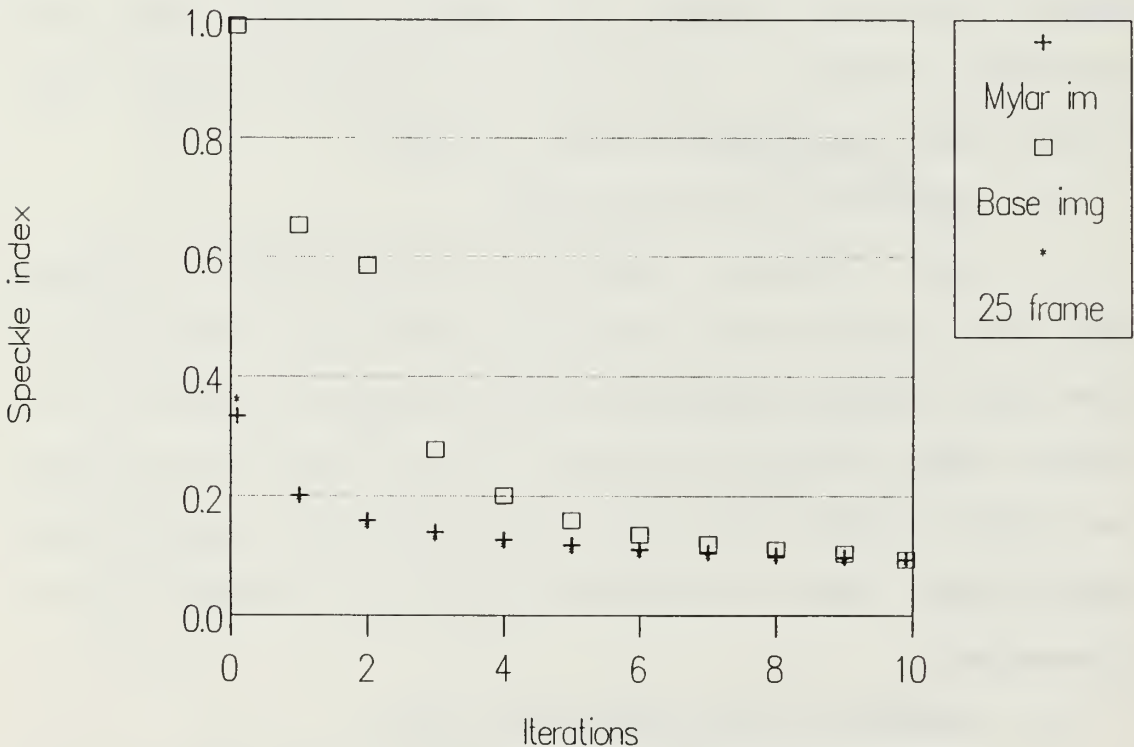
Figure 3.5 Speckle reduction by the geometric filter (the crosses are from an image recorded with a spinning mylar disk, the squares are from a non-averaged raw image, and the stars indicate data from an image averaged over 25 images).

Three comments can be made about these results. First, there is a negligible difference between the image with the mylar and the averaged frames. The geometric filter reduces the speckle a lot faster on the image without the mylar and, even though the final result is not quite as good as the others, it is still a very good result. It would be interesting to do the same measurements with the two other filters, the sigma and local statistics filters. It would also be interesting to look in more detail at the advantages and disadvantages of using only the geometric filter without any of the two averaging techniques. These questions are left for further study.

D.   SUMMARY

We described and compared two averaging techniques, aimed at reducing the speckle noise of an image. We found that the technique of applying a spinning mylar disk in the reconstruction process was more efficient at reducing the speckle index, gives a better image resolution, takes little time and requires no manipulation from the user. The only advantage of the averaging technique comes when the geometric filter is applied to a 25-frame averaged image. In this case the speckle index is slightly smaller than when applied to the image with spinning mylar. We found that a combination of both has no advantages.

Since the spinning mylar was found to be better in all cases except one, where the difference is very negligible, it is concluded that the best of the two averaging techniques is overwhelmingly the spinning mylar one. We recommend that it be used from now on in all further data collection.

IV.   STATISTIC OF PARTICLE DISTRIBUTION

The ultimate goal of this research is to be able to calculate as quickly as possible the particle distribution in the combustion chamber of a rocket motor from the hologram images recorded while firing. Two choices are offered to us, either we find the distribution of all the particles in the image or we use a minimum number of particles required to give a proper representation of the particle distribution. It is naturally more practical to make the second choice.

This chapter finds the approximate number of particles that we require in order to be confident that we have a proper representation of the particle distribution. We begin by looking at the different types of representations that we can use for the particle distribution. We will also have a look at the possible statistical tests that can be applied in our case. The Kolmogorov-Smirnov test is described and an example of its application is also given. Finally the test is applied on data collected from real images of the combustion chamber and the required number of particles is found for a given confidence level.

A.   REPRESENTATION AND STATISTICAL TEST OF SAMPLES

We decided to take samples of the sizes of the particles to find the size distribution. This means that we are now dealing with a statistical experiment. In our experiment we gather a sequence of observations (samples of the particle sizes, horizontal, vertical coordinates, and area) which can be represented many different ways. The two main forms of representation are tabular and graphical, but within those forms the information format can vary. The samples can be tabulated as they are counted, as shown in Appendix A. They also can be listed by the number of times they occur within the samples. This is called a frequency tabulation (or absolute frequency), and, if each of these values is divided by the total number of samples, we get a relative frequency tabulation. Naturally all of these tables can be represented graphically; for example, we chose for our ultimate results to represent the particle distribution by its frequency histogram, which is easily done using STATGRAPHICS.

Now that we know how to represent our data, we must be able to apply a test that will tell us how reliable a distribution we have. What we want to do is called hypothesis testing. A statistical hypothesis is an assumption about the distribution of a random variable. Examples of hypothesis includes such statements as:

1. The particle distribution is a normal distribution.

2. The particle distribution of sample A is the same as sample B.

3. The particle distribution of sample A has the same mean as sample B.

4. The unknown distribution function is a normal distribution.

Hypothesis testing is the process in which a sample is used to find out whether we may accept the hypothesis (i.e., act as though it is true) or whether we should reject it (i.e., act as though it is false). In each case the hypothesis is tested on the evidence contained in each sample. Numerous statistical tests exist (some may be found in Reference 11), and each one can be applied depending on the test data and the type of statistic we are dealing with.

B.   STATISTICS OF THE KOLMOGOROV - SMIRNOV TYPE

In our application we are trying to compare two independent samples of the particle distribution. One sample will have a larger number of particle measurements than the other sample. We want to find out if the two samples have the "same" distribution function. Our hypothesis could be worded like this: "Sample A has the same distribution function as sample B". This type of hypothesis can be tested with what is called a test for two independent samples, that is, with a test designed to compare the sample obtained (in our case the second sample,

43

with more particles), with the type of sample one would expect from the hypothesized distribution (in our case the first sample with known distribution), to see if the distribution function of the second sample "fits" the data of the first one. Tests like the median Mann - Whitney, parametric t, and Smirnov, all found in reference 11, are appropriate for this type of hypothesis. These are all nonparametric tests, meaning that they are used for non-normal distributions. Looking at the shape of the particle distribution provided by Orguc, we can see that we do not have a normal (or Gaussian) density function. A normal density function is symmetric around its mean, which is not true for our particle size distribution.

We opted for the Smirnov test, also called the Kolmogorov - Smirnov two-sample test. Conover [Ref. 11] dedicates a complete chapter on the statistics of the Kolmogorov - Smirnov type. These statistics are determined by the vertical distance between distribution functions, either hypothesized or empirical distribution functions. The Smirnov test deals with two empirical distribution functions, which is exactly what we need. The next section covers this test in more details.

C.   KOLMOGOROV - SMIRNOV TWO-SAMPLE TEST

The Kolmogorov - Smirnov two-sample test applies the Kolmogorov's statistic to two empirical distribution

functions. The Kolmogorov statistic, which we may call Tl,
is defined as the largest vertical distance between an
hypothesized and an empirical distribution function.
Applied to two empirical distribution functions, Tl can be
represented as shown in Figure 4.1. The main quality of
this test is that it is consistent against all types of
differences that may exist between the two distributions,
and that it does not measure only the difference in mean or
median for examples.
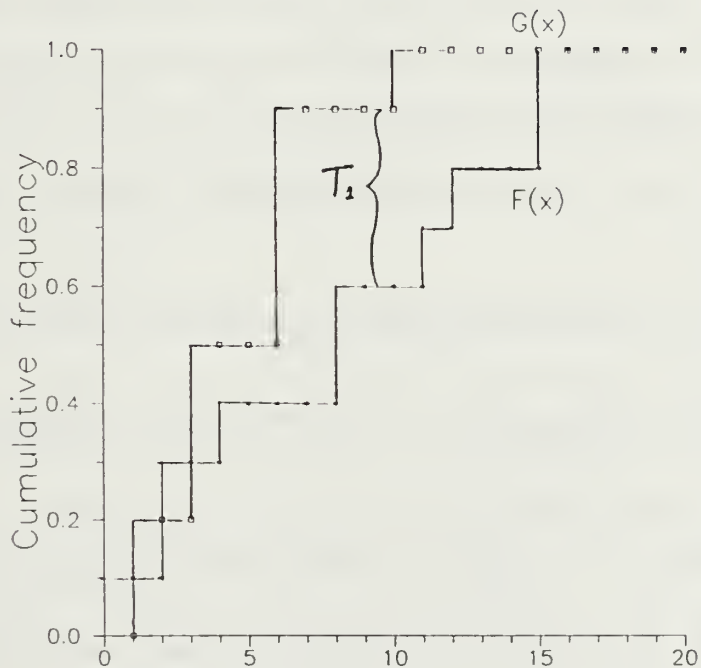
Figure 4.1   Two empirical distribution functions and
             Kolmogorov's statistic Tl.

1. Test statements

The data consist of two independent random samples, one of size n, X1, X2, ... Xn, and the other of size m, Y1, Y2, ... Ym. F(x) and G(x) are their respective unknown distribution functions.

The following assumptions are made:

1. The samples are random samples.

2. The two samples are mutually independent.

3. The measurement scale is at least ordinal.

4. For discrete variables the test is still valid but becomes more conservative than if we have continuous variables.

Once the assumptions are accepted, an hypothesis is worded, the test is conducted on the data and a decision is made on whether we reject or accept the hypothesis. Three different hypothesis or tests can be done, a two-sided test and two one-sided tests. We will use the two-sided test.

The hypothesis may be defined as follows

H0: F(x) = G(x) for all x from minus infinity to infinity

H1: F(x) ≠ G(x) for at least one value of x.

We now have to define the test statistic T1. Let S1(x) be the empirical distribution function based on the random sample X1, X2,.... Xn, and let S2(x) be the empirical distribution function based on the other random

sample Y1, Y2,.... Ym. We want to define T1, as being the greatest vertical distance between the two empirical distribution functions or

   T1 = SUP | S1(x) - S2(x) |

where "SUP | |" stands for the maximum absolute value.

Now that we have everything stated, let us define a decision rule. We will reject H0 at the level of significance A, if the appropriate test statistic T1 exceeds its 1 - A quantile as given in Appendix D. If the sample size is too large and not covered in Appendix D, the large sample approximation given at the end of the table can be used. This test permits us to compare two sample distributions and it will tell us if they are the same with a maximum error of plus or minus 2%, for example (quantile of 0.98, or p = 0.98 in Appendix D).

   2. Test example

This example of the Smirnov test is taken from reference 11. Table 4.1 gives us the values for a random sample of size nine, X1,... X9, and another one of size 15, Y1,. .. Y15. We want to know if the two populations have the same distribution functions, defined respectively as F(x) and G(x).

TABLE 4.1

TWO RANDOM SAMPLES AND THEIR EMPIRICAL
DISTRIBUTION FUNCTIONS [REF. 11].

X_i	Y_i	$S_1(x) - S_2(x)$	X_i	Y_i	$S_1(x) - S_2(x)$
	5.2	$0 - 1/15 = -1/15$		9.8	$5/9 - 8/15 = 1/45$
	5.7	$0 - 2/15 = -2/15$	9.9		$6/9 - 8/15 = 2/15$
	5.9	$0 - 3/15 = -1/5$	10.1		$7/9 - 8/15 = 11/45$
	6.5	$0 - 4/15 = -4/15$	10.6		$8/9 - 8/15 = 16/45$
	6.8	$0 - 5/15 = -1/3$		10.8	$8/9 - 9/15 = 13/45$
7.6		$1/9 - 5/15 = -2/9$	11.2		$1 - 9/15 = 2/5$
	8.2	$1/9 - 6/15 = -13/45$		11.3	$1 - 10/15 = 1/3$
8.4		$2/9 - 6/15 = -8/45$		11.5	$1 - 11/15 = 4/15$
8.6		$3/9 - 6/15 = -1/15$		12.3	$1 - 12/15 = 1/5$
8.7		$4/9 - 6/15 = 2/45$		12.5	$1 - 13/15 = 2/15$
	9.1	$4/9 - 7/15 = -1/45$		13.4	$1 - 14/15 = 1/15$
9.3		$5/9 - 7/15 = 4/45$		14.6	$1 - 1 = 0$

The two hypothesis may be written as follows

HO: F(x) = G(x) for all x from minus infinity to
     infinity

which is the hypothesis that the two distribution functions
are the same and,

H1: F(x) ≠ G(x) for at least one value of x,

the hypothesis that they are not identical.

The test statistic is given by

T1 = SUP | S1 (x) - S2 (x)|

   = 2/5 = 0.400

We see the largest absolute difference between S1(x) and S2(x) is 0.400 and occurs between x = 11.2 and x = 11.3. For this two-sided test we have, n = 9 = N1, and m = 15 = N2, and from Appendix D we see that the .95 quantile of T1 is given as W.95 = 8/15 = 0.533. In this example, T1 equals 2/5 = 0.400. Since T1 is less than the quantile value, H0 is accepted at the .05 level of significance.

Now that we know what the Smirnov test is, we will apply it to our samples in the next section.

D.   TEST APPLICATION

Based on the observation of the data distribution Orguc [Ref. 4] intuitively suggested that 1000 particles is probably a reasonable number to give us a good representation of the particle distribution. From that statement we decided to apply the Kolmogorov - Smirnov two sample test to samples with 909 and 1059 particles to start with. From these, we only have to increase the number of particles until we are satisfied with the results.

Originally the STATGRAPHICS program was supposed to be used since it has the capability to perform the test. Unfortunately due to bugs in the program, we were unable to use it. Fortunately the program could be used to calculate the cumulative relative frequency (CRF) and also plot the cumulative distribution functions (CDF). Once the CRF

49

table was produced, the largest difference between S1(x) and S2(x) was calculated using a spreadsheet program.

The first test, performed to the samples of 909 and 1054 particles, gave the following statistic T1

$$T1 = SUP \mid S1(x) - S2(x) \mid = 0.142$$

and since N1 = 909 and N2 = 1059, we used the large sample approximation which gives us the .95 quantile as

$$W.95 = 0.061$$

Taking 0.05 as a reasonable level of significance, we therefore conclude that Ho is rejected at the 0.05 level of significance. We can also see that the significance level would be very large, past the maximum 0.20 level that the table gives us. (Note that large significance values are undesirable.) We see from this test that the two distributions are fairly different. Figure 4.2 shows the two CDF's for the 909 and 1059 particle samples. As we can see, just looking at the plots does not permit us to decide whether or not the two distributions are the same. From the plots we would tend to say they are very close to one another, but the test proves otherwise. We can see that the test is definitely required.

The next test was performed on two samples of 1059 and 1179 particles. The statistical test was found to be

$$T1 = SUP \mid S1(x) - S2(x) \mid = 0.020$$

Since the N1 = 1059 and N2 = 1179 values give .95 quantile as

$$W.95 = 0.058$$

We, therefore, accept Ho at the .05 level of significance. This is to say that the two distribution functions may be considered to be identical within the 5% confidence level.

From the test we can conclude that after a count of about 1100 particles, the variation of the particle distribution should be negligible, and therefore this latest distribution can be considered as a good representation of the particles distribution.

Before we go to the next section let us make a few comments about the test we just performed. The test results should not be taken as absolute truth for two reasons. First, the assumption that we have two independent samples is not necessarily true because we used the same hologram to extract the samples, so some particles may be counted in both samples. The same test should be done using samples of different holograms. The second reason for not taking this test for granted is that the results obtained are good for the samples used, and nothing guarantees us that these two samples have representative distributions for all firing rocket motors. This test

tells us that for the hologram that we used, a stable distribution is obtained after about 1100 particles are counted. A lot more study remains to be done on the statistic of the particle distribution. The test should be performed on several different hologram images.

A level of significance of 0.05 was arbitraily chosen as been a reasonable value. It should be noted that a different value could be much better suited for this problem. Further study should be conducted to find the optimum level of significance in this case.
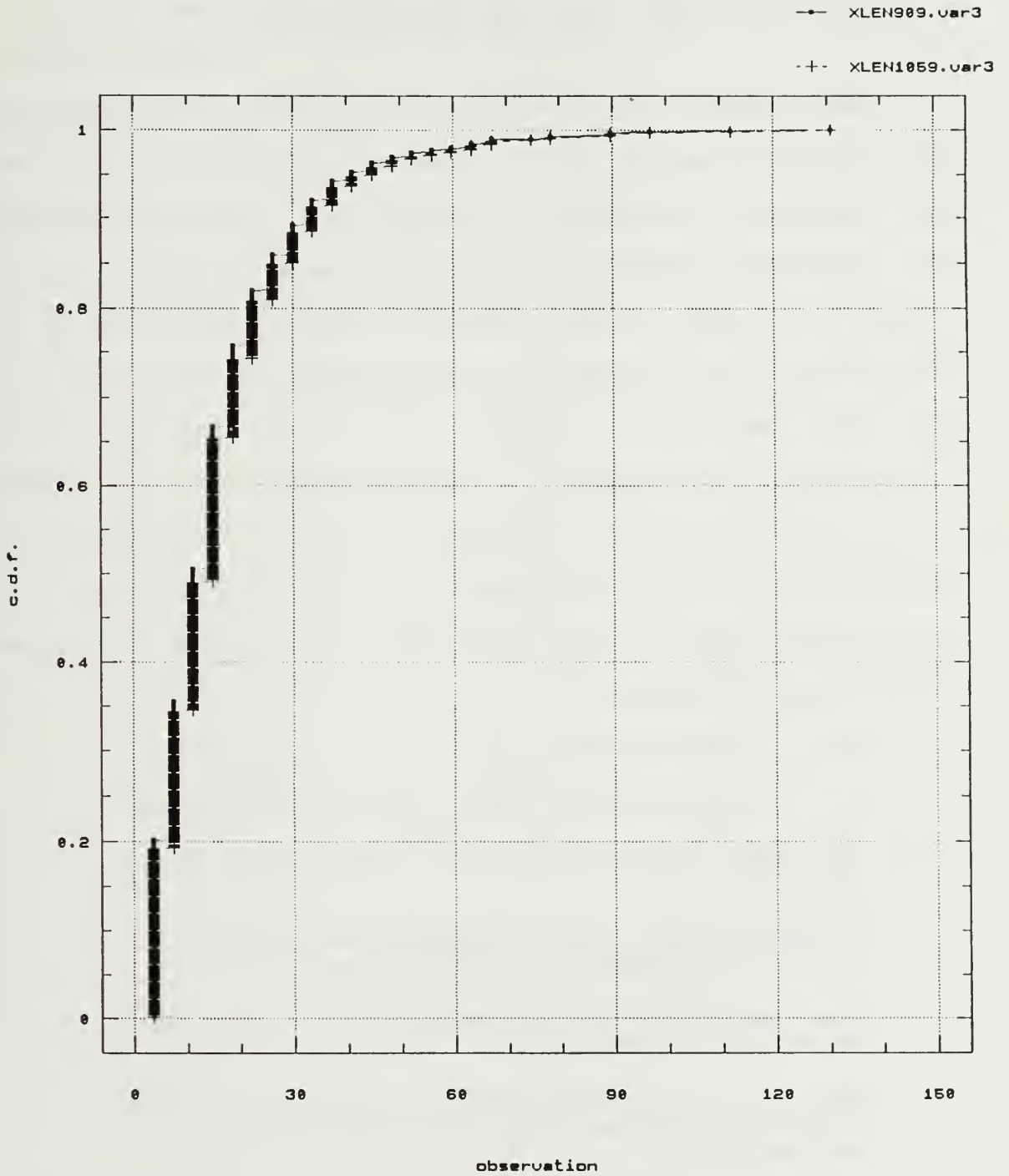
Figure 4.2 CDF's for two samples, 909 and 1059
           particles.

V. PARTICLE DISTRIBUTIONS

This chapter contains the final product obtained from the research done so far. Figures 5.1, 5.2 and 5.3 show the frequency histograms of the particle distributions in the combustion chamber of a rocket motor while firing. Figure 5.1 gives us the particle size distribution as a function of the X length (or X coordinate) of each particle for 1179 particles. Figure 5.2 is the same but as a function of the Y length. Finally Figure 5.3 is for the area of the particles. Appendix E gives us the frequency tabulation of the X coordinate. It is not our intention to analyze the data but the following observations were made on the results obtained:

1. The X length varies from 3 to 173 micrometers.

2. The Y length varies from 3 to 188 micrometers.

3. The area varies from 13 to 12818 micrometers squared.

4. Some particles were not spherical as seen by the difference in X and Y lengths.

5. The particle size is measured in discrete sizes with gaps in between.

6. The frequency histograms are showing particles below the smallest possible resolution of 14 micrometers.

A detailed analysis of the particle distribution should be done with the data available from several

different good holograms of the combustion chamber. Such
holograms were just becoming available when this report was
being finished.
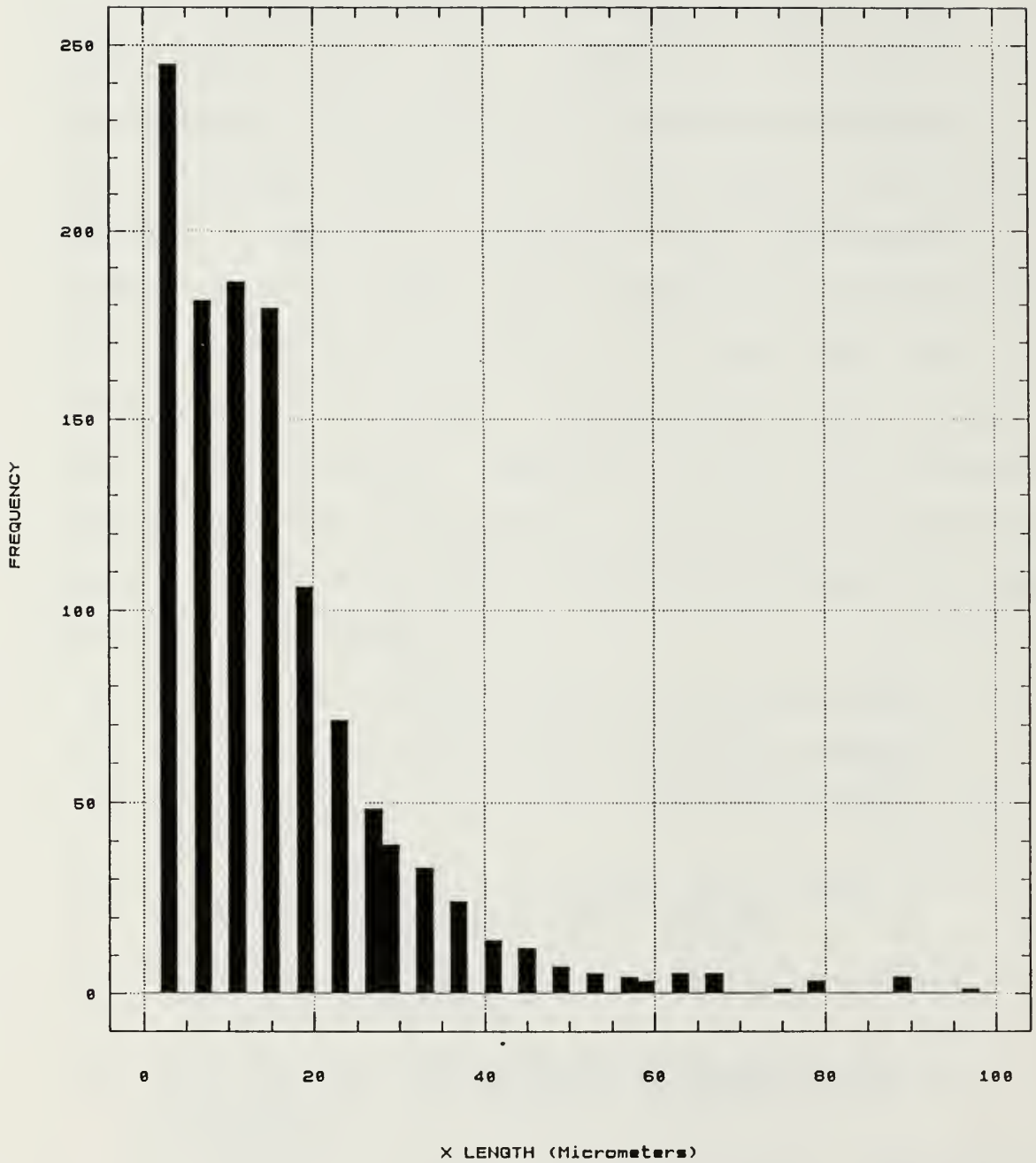
Figure 5.1   Particle frequency histogram
(X length, 1179 particles)

56
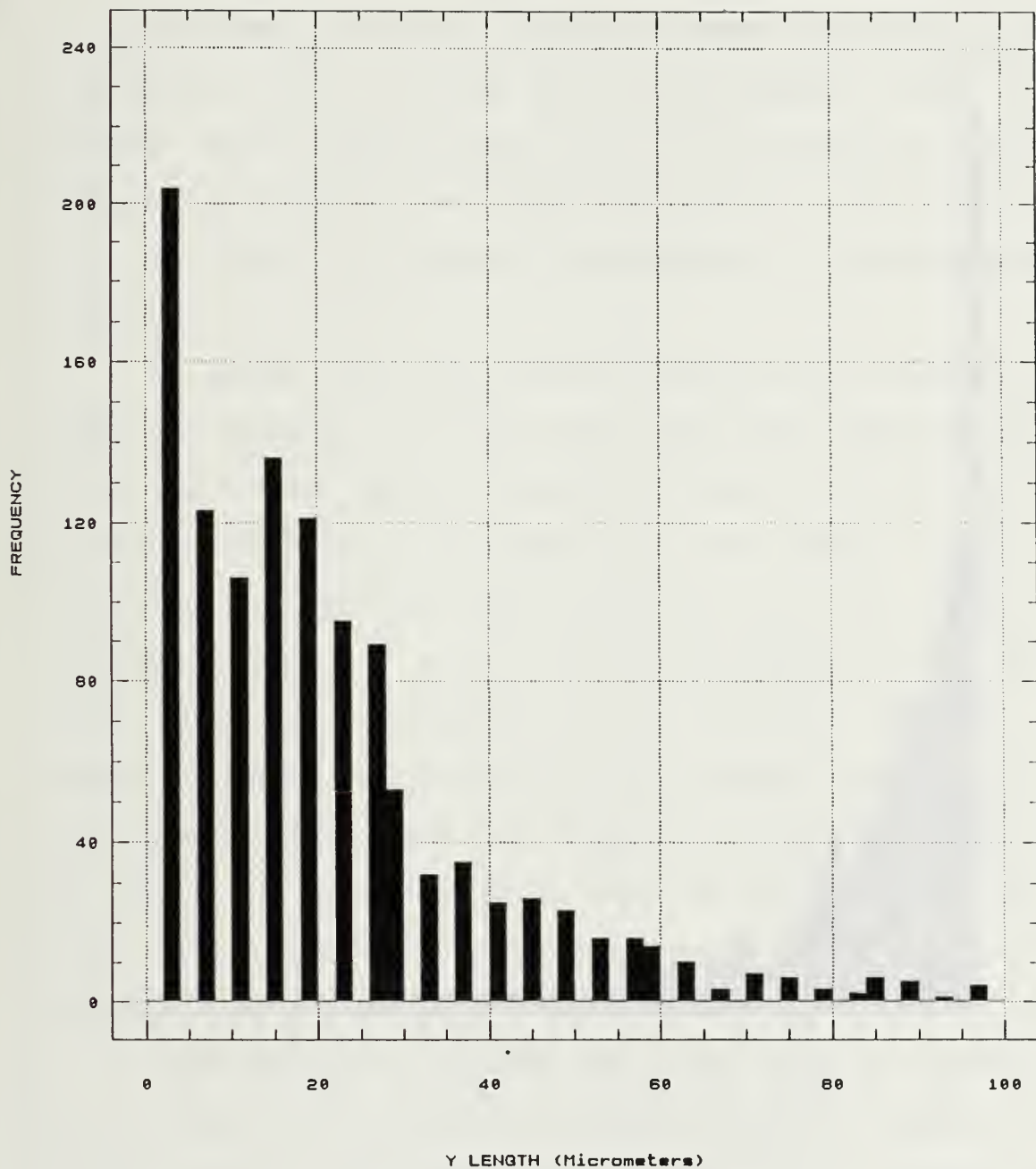
Figure 5.2   Particle frequency histogram
              (Y length, 1179 particles)
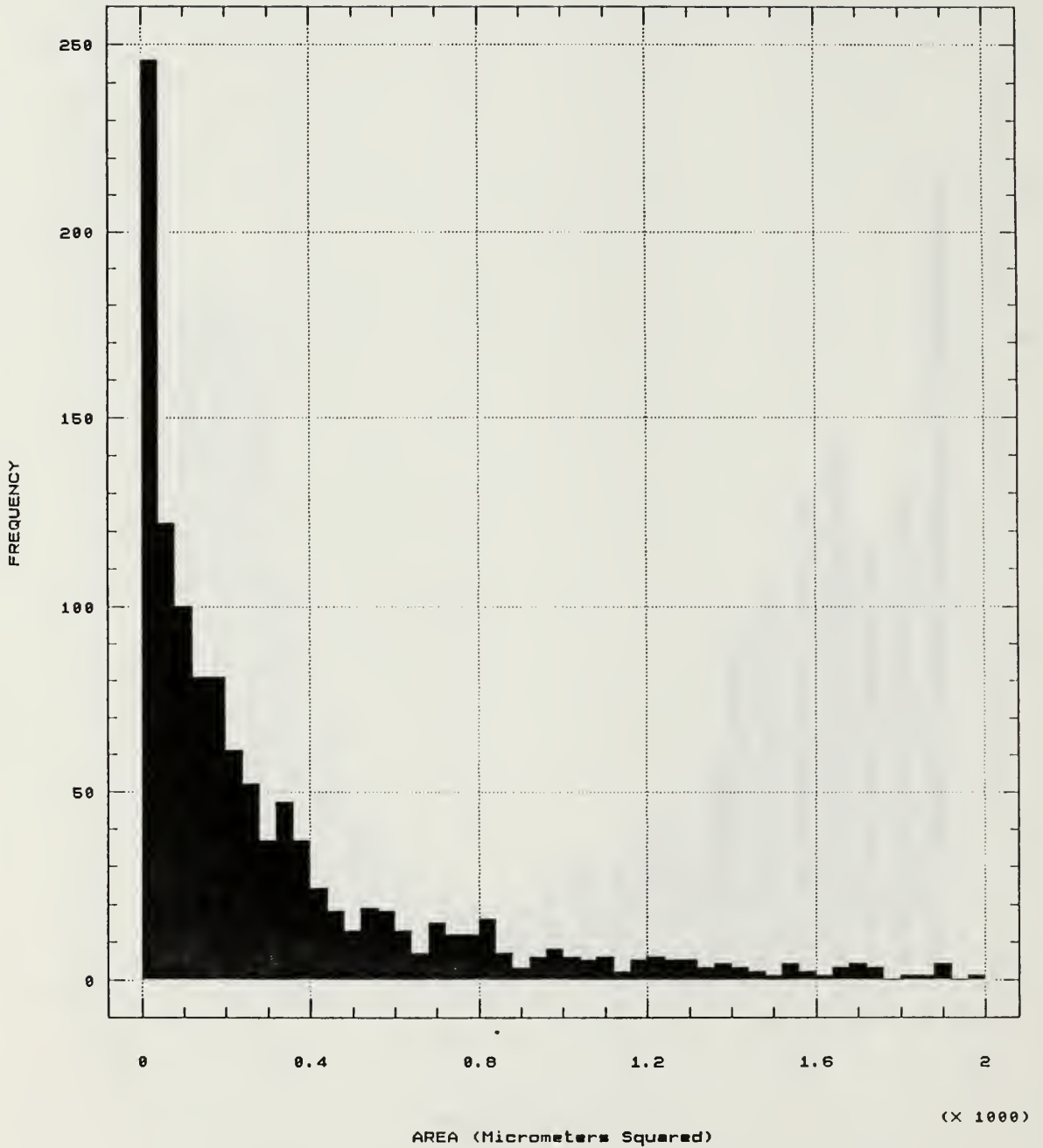
Figure 5.3 Particle frequency histogram
(area, 1179 particle)

VI. CONCLUSIONS

The results of this thesis continue to show that the IBM PC/AT with a 16 MHz 386 microprocessor board, a PC Vision frame grabber board, the Itex PC software package and various locally developed programs is a very reliable way to extract the particle distribution in the combustion chamber.

We think that the speckle reduction has reached its optimum capacity. It was found that the spinning mylar disk was a much better technique to use than the software averaging programs developed in this report. It is therefore recommended that the mylar disk be used for further research. It was very interesting to see that, after about ten iterations of the geometric filter, the speckle index was the same for the images obtained using the mylar disk, 25 averaged frames, and the image with none of those two. Further study could be done to investigate this interesting observation, since there could be advantages of not averaging the speckle noise.

The Kolmogorov - Smirnov two sample test was performed on a sample of particles in the combustion chamber of a small rocket motor while firing, and it was found that about 1100 particles are required to obtain a stable particle distribution with a level of significance of 0.05.

The test was performed on only one hologram image but later should be done on several different hologram images to confirm its validity. An optimum level of significance should be investigated for our problem.

STATGRAPHICS turned out to be a very unreliable program due to bugs in it. It is recommended that another program be used for any statistical related matter, unless the company makes modifications to it.

The hologram quality still need to be improved along with the hologram reconstruction process.

The next step to improve the speed of the whole process is to write the routines in "C" language, and it is currently been done. A new COMPAQ micro-computer is now available along with an ImageAction Plus package which should help increase the speed.

The automatic data extraction process is now in very good shape and we are ready to gather more data for analysis purposes.

APPENDIX A

SAMPLE FILE OF PARTICLE SIZING PROGRAM

The first column is the particle
identification number, the second one
is its area, the third is its x length,
and the fourth is its y length.

1	111.6194	11.2059	11.2059
2	1353.3850	41.0883	37.3530
3	795.2882	33.6177	33.6177
4	404.6203	18.6765	26.1471
5	1883.5770	44.8236	63.5001
6	1967.2920	41.0883	74.7060
7	892.9552	33.6177	33.6177
8	265.0961	14.9412	22.4118
9	572.0494	22.4118	33.6177
10	265.0961	14.9412	22.4118
11	460.4300	18.6765	29.8824
12	390.6679	22.4118	26.1471
13	139.5242	11.2059	14.9412
14	990.6222	33.6177	41.0883
15	69.7621	11.2059	7.4706
16	167.4291	14.9412	14.9412
17	948.7649	37.3530	33.6177
18	404.6203	18.6765	26.1471
19	376.7155	22.4118	22.4118
20	795.2882	33.6177	33.6177
21	1618.4810	44.8236	56.0295
22	3432.2970	67.2354	67.2354
23	111.6194	11.2059	11.2059
24	809.2407	33.6177	33.6177
25	69.7621	11.2059	7.4706
26	167.4291	11.2059	22.4118
27	1409.1950	26.1471	67.2354
28	69.7621	11.2059	7.4706
29	613.9067	26.1471	33.6177
30	13.9524	3.7353	3.7353
31	390.6679	22.4118	26.1471
32	1102.2420	41.0883	33.6177
33	13.9524	3.7353	3.7353
34	97.6670	11.2059	11.2059
35	111.6194	11.2059	14.9412
36	55.8097	7.4706.	7.4706
37	320.9058	18.6765	22.4118
38	1046.4320	33.6177	41.0883
39	139.5242	14.9412	11.2059
40	1451.0520	41.0883	44.8236
41	1046.4320	37.3530	33.6177
42	725.5261	29.8824	33.6177
43	27.9049	3.7353	7.4706
44	1492.9090	56.0295	44.8236
45	892.9552	33.6177	33.6177
46	1632.4340	41.0883	63.5001
47	892.9552	33.6177	33.6177
48	572.0494	22.4118	33.6177
49	446.4776	18.6765	29.8824

50	390.6679	22.4118	26.1471
51	446.4776	18.6765	29.8824
52	1185.9560	37.3530	41.0883
53	1032.4790	37.3530	41.0883
54	990.6222	33.6177	41.0883
55	1297.5760	44.8236	33.6177
56	795.2882	33.6177	33.6177
57	1032.4790	37.3530	41.0883
58	1451.0520	41.0883	52.2942
59	1674.2910	44!8236	59.7648
60	1841.7200	44.8236	63.5001
61	460.4300	26.1471	26.1471
62	1032.4790	37.3530	41.0883
63	69.7621	11.2059	7.4706
64	209.2864	14.9412	18.6765
65	460.4300	26.1471	26.1471
66	558.0970	26.1471	26.1471
67	390.6679	22.4118	26.1471
68	362.7631	14.9412	29.8824
69	125.5718	14.9412	11.2059
70	55.8097	7.4706	7.4706
71	55.8097	11.2059	7.4706
72	558.0970	26.1471	26.1471
73	279.0485	14.9412	22.4118
74	362.7631	14.9412	29.8824
75	55.8097	7.4706	7.4706
76	1046.4320	33.6177	41.0883
77	1702.1960	48.5589	44.8236
78	55.8097	11.2059	7.4706
79	1451.0520	41.0883	44.8236
80	558.0970	26.1471	29.8824
81	27.9049	3.7353	7.4706
82	739.4785	33.6177	26.1471
83	460.4300	26.1471	26.1471
84	1088.2890	26.1471	48.5589
85	460.4300	26.1471	26.1471
86	97.6670	11.2059	11.2059
87	237.1912	14.9412	18.6765
88	55.8097	7.4706	11.2059
89	3250.9150	56.0295	74.7060
90	181.3815	18.6765	11.2059
91	739.4785	33.6177	26.1471
92	69.7621	7.4706	11.2059
93	1185.9560	37.3530	41.0883
94	1841.7200	44.8236	63.5001
95	3264.8670	56.0295	74.7060
96	823.1931	29.8824	37.3530
97	1032.4790	37.3530	41.0883
98	251.1436	14.9412	18.6765
99	139.5242	37.3530	3.7353
100	111.6194	29.8824	3.7353
101	55.8097	14.9412	3.7353

APPENDIX B

PROGRAM TO DIVIDE IMAGE FRAMES IN HALVES

```
$STORAGE:2
$NOLIST
$INCLUDE:'ITEXPC.INC'
$LIST
      PROGRAM SFAVGA

***********************************************************************
*                                                                     *
*       THIS PROGRAM SAVES N IMAGES INTO N FILES READY FOR THE PROGRAM *
*       AVGA . EACH IMAGE FILE IS SAVED AS TWO FILES SUCH THAT THE     *
*       LEFT HALF OF THE IMAGE IS STORED AS 'LHS01.IMG', AND THE RIGHT *
*       HALF OF THE IMAGE IS SAVED AS 'RHS01.IMG', FOR IMG01.IMG , ETC... *
*                                                                     *
*       THE IMAGE FRAME FILES MUST ALREADY BE STORED AS IMG01.IMG ,ETC... *
*                                                                     *
*       IT USES SUBROUTINES IMREAD , IMSAVE & SUBPROGRAM BLOCK DATA TOTAL.*
*                                                                     *
*       IT WILL ASK YOU FOR:                                          *
*          1. THE NUMBER OF IMAGES TO AVERAGE (NOI)                   *
*          2. THE COMPRESS VALUE (0 OR 1)                             *
*                                                                     *
*       NO COMMENT CAN BE ADDED TO THE SAVED IMAGES.                  *
*                                                                     *
*       DENIS J.G. CARRIER , NPS MONTEREY CA, 03 AUG 88               *
*                                                                     *
***********************************************************************

******* DECLARE VARIABLES **********************************************

      IMPLICIT INTEGER*2 (A-Z)
      COMMON /AOI/ IXS,IYS,NCOL,NROW,C,COMFLG,COMLIN
      COMMON /DEF/ LS(25),RS(25),VI(25)
      INTEGER*2 READFT
      CHARACTER*21 FNAME
      CHARACTER*127 COMLINE
      CHARACTER*11 LS,RS,VI

******* DECLARE HARDWARE JUMPER SETTINGS, INITIALIZE & CLEAR SCREEN *******

      CALL SETHDW(16#FF00, 16#D000,  1)
      CALL INITIA
      CALL SCLEAR(100,1)


***** START THE PROGRAM *************************************************

      WRITE(*,*) ' HOW MANY IMAGES TO AVERAGE --->'
      READ(*,*) NOI
      WRITE(*,'(A\)') ' ENTER THE COMPRESS VALUE (0/1): '
      READ (*,*) COMFLG

      DO 10 C = 1,NOI
```

63

```
******* SET AOI TO FULL SCREEN ***********************************************

        IXS = 0
        IYS = 0
        NROW = 480
        NCOL = 512

          CALL IMREAD

          CALL IMSAVE

10      CONTINUE

        WRITE(*,'(A\,I3,A\)') ' YOU HAVE NOW SAVED',NOI,' IMAGES !'

        STOP
        END

        SUBROUTINE IMREAD

*************** SUBROUTINE IMREAD  ***************************
*                                                                *
*         USE WITH ITEXPC TO READ AN IMAGE FROM DISK.            *
*                                                                *
******************************************************************

*     ****  DECLARATIONS *****************************************

        IMPLICIT INTEGER*2 (A-Z)
        INTEGER*2 READFT
        CHARACTER*21 FNAME
        CHARACTER*127 COMLIN
        CHARACTER*11 LS,RS,VI

*     ****READ AOI IN COMMON ****

        COMMON /AOI/ IXS,IYS,NCOL,NROW,C,COMFLG,COMLIN
        COMMON /DEF/ LS(25),RS(25),VI(25)

*     **** READ THE IMAGE ****

54      WRITE(*,*) '*** READFT ***'
        WRITE(*,*) '...READING INTO AOI...'
        WRITE(*,*) ' REMINDER: COMMENT LINES MUST END WITH A  !.'
        FNAME = VI(C)
        ERRVAL=READFT(IXS,IYS,NCOL,NROW,FNAME,COMLINE)

*     ******* READFT ERROR HANDLER *********

        WRITE(*,*) 'ERROR CODE FROM READFT = ',ERRVAL
        IF (ERRVAL.EQ.0) THEN
           WRITE(*,*) 'SUCCESSFUL FILE READ.'
           WRITE(*,*) COMLIN
         ELSEIF ((ERRVAL.EQ.-2).OR.(ERRVAL.EQ.98)) THEN
           WRITE(*,*) 'FILE NOT FOUND. TRY AGAIN.'
           GOTO 54
         ELSEIF (ERRVAL.EQ.-3) THEN
           WRITE(*,*) 'THE FILE FORMAT IS BAD. TRY ANOTHER FILE NAME.'
           GOTO 54
         ELSEIF (ERRVAL.EQ.-7) THEN
           WRITE(*,*) 'ERROR MADE IN ENTERING FILENAME. TRY AGAIN.'
GOTO 54
        ENDIF
        END
```

64

```
***** END OF IMREAD ************************************************************
      SUBROUTINE IMSAVE

********************* SUBROUTINE IMSAVE ****************************************
*                                                                             *
*     USE TO SAVE THE AOI TO DISK.                                            *
*                                                                             *
******************************************************************************

***** DECLARATIONS ************************************************************

      IMPLICIT INTEGER*2 (A-Z)
      INTEGER*2 IXS,IYS,NCOL,NROW,COMFLG
      CHARACTER*21 FNAME
      CHARACTER*127 COMLIN
      CHARACTER*11 LS,RS,VI

      COMMON /AOI/ IXS,IYS,NCOL,NROW,C,COMFLG,COMLIN '
      COMMON /DEF/ LS(25),RS(25),VI(25)

*    **** PERFORM SAVE OPERATION. ****

      CALL SETHDW(16#FF00, 16#D000,  1)
      CALL INITIA
65    WRITE(*,*) '*** SAVEFT ***'
      WRITE(*,*) '...HALF SCREEN SAVE...'
      DO 30 D=1,2
        IF(D.EQ.2)GOTO 100
          FNAME = LS(C)
          GOTO 110
100   FNAME = RS(C)
110   NCOL=256
        IF(D.EQ.1)GOTO 120
          IXS=256
120   ERRVAL=SAVEFT(IXS,IYS,NCOL,NROW,COMFLG,FNAME,COMLIN)

*        ****** SAVEFT ERROR HANDLER ********

      IF (ERRVAL.EQ.0) THEN
          WRITE(*,'(A\,I2,A\)') ' HALF OF FILE IMG.',C,
     $                          ' WRITTEN SUCCESSFULLY '
      WRITE(*,*) ' '
        ELSEIF (ERRVAL.EQ.-1) THEN
          WRITE(*,*) 'INSUFFICIENT DISK SPACE.'
          GOTO 65
        ELSEIF (ERRVAL.EQ.-7) THEN
          WRITE(*,*) 'FILENAME ERROR.'
          GOTO 65
        ELSEIF (ERRVAL.EQ.-8) THEN
          WRITE(*,*) 'COMMENT LINE ERROR'
          GOTO 65
        ELSE
          GOTO 65
      ENDIF
30    CONTINUE

      END

********************* END OF IMSAVE ********************************
```

```
************************************************************************
*                                                                      *
*       THIS BLOCK DATA DEFINES VARIABLES LS & RS WHICH ARE THE        *
*       THE NAMES OF THE FILES TO RETRIEVE AND VI THE FILE TO BE SAVED. *
*                                                                      *
************************************************************************

      BLOCK DATA TOTAL
      COMMON /DEF/ LS(25),RS(25),VI(25)
      CHARACTER*11 LS,RS,VI

      DATA LS /'D:LHS01.IMG','D:LHS02.IMG','D:LHS03.IMG',
     $ 'D:LHS04.IMG','D:LHS05.IMG','D:LHS06.IMG','D:LHS07.IMG',
     $ 'D:LHS08.IMG','D:LHS09.IMG','D:LHS10.IMG','D:LHS11.IMG',
     $ 'D:LHS12.IMG','D:LHS13.IMG','D:LHS14.IMG','D:LHS15.IMG',
     $ 'D:LHS16.IMG','D:LHS17.IMG','D:LHS18.IMG','D:LHS19.IMG',
     $ 'D:LHS20.IMG','D:LHS21.IMG','D:LHS22.IMG','D:LHS23.IMG',
     $ 'D:LHS24.IMG','D:LHS25.IMG'/

      DATA RS /'D:RHS01.IMG','D:RHS02.IMG','D:RHS03.IMG',
     $ 'D:RHS04.IMG','D:RHS05.IMG','D:RHS06.IMG','D:RHS07.IMG',
     $ 'D:RHS08.IMG','D:RHS09.IMG','D:RHS10.IMG','D:RHS11.IMG',
     $ 'D:RHS12.IMG','D:RHS13.IMG','D:RHS14.IMG','D:RHS15.IMG',
     $ 'D:RHS16.IMG','D:RHS17.IMG','D:RHS18.IMG','D:RHS19.IMG',
     $ 'D:RHS20.IMG','D:RHS21.IMG','D:RHS22.IMG','D:RHS23.IMG',
     $ 'D:RHS24.IMG','D:RHS25.IMG'/

      DATA VI /'D:IMG01.IMG','D:IMG02.IMG','D:IMG03.IMG',
     $ 'D:IMG04.IMG','D:IMG05.IMG','D:IMG06.IMG','D:IMG07.IMG',
     $ 'D:IMG08.IMG','D:IMG09.IMG','D:IMG10.IMG','D:IMG11.IMG',
     $ 'D:IMG12.IMG','D:IMG13.IMG','D:IMG14.IMG','D:IMG15.IMG',
     $ 'D:IMG16.IMG','D:IMG17.IMG','D:IMG18.IMG','D:IMG19.IMG',
     $ 'D:IMG20.IMG','D:IMG21.IMG','D:IMG22.IMG','D:IMG23.IMG',
     $ 'D:IMG24.IMG','D:IMG25.IMG'/

      END
```

PROGRAM TO AVERAGE SEVERAL IMAGE FRAMES


```
$STORAGE:2
$NOLIST
$INCLUDE:'ITEXPC.INC'
$LIST
      PROGRAM AVGA

**********************************************************************
*     THIS PROGRAM  AVERAGES N IMAGES AND OUTPUT THE RESULT IN THE FILE  *
*     AVGIXX.IMG , WHERE XX IS THE NUMBER OF IMAGES AVERAGED .           *
*                                                                       *
*     IT USES THE FILES CREATED BY THE PROGRAM SFAVGA .                 *
*     IT IS NOW DIMENSIONED TO AVERAGE UP TO 25 IMAGES .                *
*                                                                       *
*     DENIS J.G. CARRIER , NPS MONTEREY CA , 03 AUG 88 .                *
**********************************************************************

***** DECLARE VARIABLES ***********************************************

      IMPLICIT INTEGER*2 (A-Z)
      INTEGER*2 READFT,SAVEFT,X,Y,DATUM
      INTEGER*2 XS,YS,DXS,DYS,XD,YD,DXD,DYD,MODE,OUTFLAG
      COMMON /AOI/ IXS,IYS,NROW,NCOL,A,B
      COMMON /DEF/ LS(25),RS(25)
      CHARACTER*21 FNAME
      CHARACTER*127 COMLINE
      CHARACTER*11 LS,RS
      DIMENSION ARRAY(256,480)

***** DEFINE HARDWARE JUMPER SETTINGS, INITIALIZE & CLEAR SCREEN **********

      CALL SETHDW(16#FF00, 16#D000,  1)
      CALL INITIA
      CALL SCLEAR(100,1)

***** SET AOI LEFT HALF OF THE SCREEN *********************************

      IXS = 0
      IYS = 0
      NROW = 480
      NCOL = 256

***** DEFINE SOURCE & DESTINATION FOR MCAREA COMMAND *********************
***** (SOURCE-LEFT HALF OF SCREEN , DESTINATION-RIGHT HALF OF SCREEN ******

      XS = 0
      YS = 0
      DXS = 256
      DYS = 480
      XD = 256
      YD = 0
      DXD = 256
      DYD = 480
      MODE = 4
      OUTFLAG = 2
```

```
***** START OF PROGRAM **************************************************************

      WRITE(*,*) ' HOW MANY IMAGES TO AVERAGE '
      READ(*,*) NOI

      DO 10 A = 1,2

***** INITIALIZE THE ARRAY TO 0 ***************************************************

        DO 15 X = 1,256
          DO 16 Y =1,480
            ARRAY(X,Y) = 0
16        CONTINUE

15      CONTINUE

***** READ ONE HALF IMAGE  ********************************************************

        DO 20 B = 1,NOI
          CALL IMREAD

***** READ EACH PIXEL VALUE AND PLACE IT IN THE PROPER ARRAY POSITION *****
***** AND ADD THEM ALL ***********************************************************

        DO 30 I = 1,256

          DO 40 J = 1,480
            ARRAY(I,J) = ARRAY(I,J) + RPIXEL((I-1),(J-1))
40        CONTINUE

30      CONTINUE

20      CONTINUE

***** DEVIDE EACH PIXEL VALUE BY THE NUMBER OF IMAGES (NOI) ***************

        DO 50 K = 1,256

          DO 60 L = 1,480
            ARRAY(K,L) = ARRAY(K,L) / NOI
            CALL WPIXEL((K-1),(L-1),ARRAY(K,L))
60        CONTINUE

50      CONTINUE

***** IF WORKING ON RIGHT IMAGE MOVE AVERAGED VALUES TO THE RIGHT OF SCREEN

        IF(A.EQ.2) GOTO 10
          CALL QCLEAR(1,0)
          CALL QCLEAR(3,0)
          CALL MCAREA(XS,YS,DXS,DYS,XD,YD,DXD,DYD,MODE,OUTFLAG)

10      CONTINUE

      NCOL = 512

      CALL IMSAVE

      STOP
      END

      SUBROUTINE IMREAD
```

```
*************** SUBROUTINE IMREAD ***************************
*                                                          *
*       USE WITH ITEXPC TO READ AN IMAGE FROM DISK.        *
*                                                          *
*******  DECLARATIONS ***************************************

       IMPLICIT INTEGER*2 (A-Z)
       INTEGER*2 READFT
       CHARACTER*21 FNAME
       CHARACTER*127 COMLIN
       CHARACTER*11 LS,RS

*    ****READ AOI IN COMMON ****

       COMMON /AOI/ IXS,IYS,NROW,NCOL,A,B
       COMMON /DEF/ LS(25),RS(25)

*    **** READ THE IMAGE ****

54     WRITE(*,*) '*** READFT ***'
       WRITE(*,*) '...READING INTO AOI...'
       WRITE(*,*) 'REMINDER: COMMENT LINES MUST END WITH A  !.'
       IF(A.EQ.2) GOTO 57
         FNAME=RS(B)
         GOTO 58
57     FNAME=LS(B)
58     ERRVAL=READFT(IXS,IYS,NCOL,NROW,FNAME,COMLIN)

*    ******* READFT ERROR HANDLER *********

       WRITE(*,*) 'ERROR CODE FROM READFT = ',ERRVAL
       IF (ERRVAL.EQ.0) THEN
           WRITE(*,*) 'SUCCESSFUL FILE READ.'
           WRITE(*,*) COMLIN
        ELSEIF ((ERRVAL.EQ.-2).OR.(ERRVAL.EQ.98)) THEN
           WRITE(*,*) 'FILE NOT FOUND. TRY AGAIN.'
           GOTO 54
        ELSEIF (ERRVAL.EQ.-3) THEN
           WRITE(*,*) 'THE FILE FORMAT IS BAD. TRY ANOTHER FILE NAME.'
           GOTO 54
        ELSEIF (ERRVAL.EQ.-7) THEN
           WRITE(*,*) 'ERROR MADE IN ENTERING FILENAME. TRY AGAIN.'
           GOTO 54
       ENDIF

       END

       SUBROUTINE IMSAVE

********************** SUBROUTINE IMSAVE *******************************
*                                                                     *
*       USE TO SAVE THE AOI TO DISK.                                  *
*                                                                     *
**********************************************************************

*    **** DECLARATIONS ****

       IMPLICIT INTEGER*2 (A-Z)
       INTEGER*2 SAVEFT
       CHARACTER*21 FNAME
       CHARACTER*127 COMLINE
```

69

```fortran
       CHARACTER*11 LS,RS
       INTEGER*2 ERRVAL
       COMMON /AOI/ IXS,IYS,NROW,NCOL,A,B
       COMMON /DEF/ LS(25),RS(25)

*    **** PERFORM SAVE OPERATION. ****

       CALL SETHDW(16#FF00, 16#D000,  1)
       CALL INITIA
65     WRITE(*,*) '*** SAVEFT ***'
       WRITE(*,*) '...HALF SCREEN SAVE...'
       WRITE(*,'(A\,I2,A\)') ' ENTER FILENAME AS DEV:AVGI',B-1,'.IMG--->'
       READ (*,'(A)') FNAME
       WRITE(*,'(A\)') ' ENTER THE COMPRESS VALUE (0/1):'
       READ (*,*) COMFLG
       WRITE(*,*) 'ENTER THE IMAGE COMMENT, TERMINATED BY A "!" :'
       READ (*,'(A)') COMLIN
       ERRVAL=SAVEFT(IXS,IYS,NCOL,NROW,COMFLG,FNAME,COMLIN)

*        ****** SAVEFT ERROR HANDLER ********

       IF (ERRVAL.EQ.0) THEN
          WRITE(*,*) 'FILE WRITTEN SUCCESSFULLY.'
       ELSEIF (ERRVAL.EQ.-1) THEN
         WRITE(*,*) 'INSUFFICIENT DISK SPACE.'
         GOTO 65
       ELSEIF (ERRVAL.EQ.-7) THEN
         WRITE(*,*) 'FILENAME ERROR.'
         GOTO 65
       ELSEIF (ERRVAL.EQ.-8) THEN
         WRITE(*,*) 'COMMENT LINE ERROR'
         GOTO 65
       ELSE
         GOTO 65
       ENDIF

* ****************** END OF IMSAVE  ************************************

       END

**********************************************************************
*                                                                    *
*     THIS BLOCK DATA DEFINES VARIABLES LS & RS WHICH ARE THE         *
*     THE NAMES OF THE FILES TO RETRIEVE .                            *
*                                                                    *
**********************************************************************

       BLOCK DATA TOTAL
       COMMON /DEF/ LS(25),RS(25)
       CHARACTER*11 LS,RS

       DATA LS /'D:LHS01.IMG','D:LHS02.IMG','D:LHS03.IMG',
      $ 'D:LHS04.IMG','D:LHS05.IMG','D:LHS06.IMG','D:LHS07.IMG',
      $ 'D:LHS08.IMG','D:LHS09.IMG','D:LHS10.IMG','D:LHS11.IMG',
      $ 'D:LHS12.IMG','D:LHS13.IMG','D:LHS14.IMG','D:LHS15.IMG',
      $ 'D:LHS16.IMG','D:LHS17.IMG','D:LHS18.IMG','D:LHS19.IMG',
      $ 'D:LHS20.IMG','D:LHS21.IMG','D:LHS22.IMG','D:LHS23.IMG',
      $ 'D:LHS24.IMG','D:LHS25.IMG'/

       DATA RS /'D:RHS01.IMG','D:RHS02.IMG','D:RHS03.IMG',
      $ 'D:RHS04.IMG','D:RHS05.IMG','D:RHS06.IMG','D:RHS07.IMG',
      $ 'D:RHS08.IMG','D:RHS09.IMG','D:RHS10.IMG','D:RHS11.IMG',
      $ 'D:RHS12.IMG','D:RHS13.IMG','D:RHS14.IMG','D:RHS15.IMG',
      $ 'D:RHS16.IMG','D:RHS17.IMG','D:RHS18.IMG','D:RHS19.IMG',
      $ 'D:RHS20.IMG','D:RHS21.IMG','D:RHS22.IMG','D:RHS23.IMG',
      $ 'D:RHS24.IMG','D:RHS25.IMG'/

       END
```

APPENDIX D

QUANTILES OF THE SMIRNOV TEST STATISTIC FOR TWO SAMPLES (From Ref.11).

One-Sided Test: Two-Sided Test:			$p = .90$ $p = .80$	.95 .90	.975 .95	.99 .98	.995 .99
$N_1 = 1$	$N_2 =$	9	17/18				
		10	9/10				
$N_1 = 2$	$N_2 =$	3	5/6				
		4	3/4				
		5	4/5	4/5			
		6	5/6	5/6			
		7	5/7	6/7			
		8	3/4	7/8	7/8		
		9	7/9	8/9	8/9		
		10	7/10	4/5	9/10		
$N_1 = 3$	$N_2 =$	4	3/4	3/4			
		5	2/3	4/5	4/5		
		6	2/3	2/3	5/6		
		7	2/3	5/7	6/7	6/7	
		8	5/8	3/4	3/4	7/8	
		9	2/3	2/3	7/9	8/9	8/9
		10	3/5	7/10	4/5	9/10	9/10
		12	7/12	2/3	3/4	5/6	11/12
$N_1 = 4$	$N_2 =$	5	3/5	3/4	4/5	4/5	
		6	7/12	2/3	3/4	5/6	5/6
		7	17/28	5/7	3/4	6/7	6/7
		8	5/8	5/8	3/4	7/8	7/8
		9	5/9	2/3	3/4	7/9	8/9
		10	11/20	13/20	7/10	4/5	4/5
		12	7/12	2/3	2/3	3/4	5/6
		16	9/16	5/8	11/16	3/4	13/16
$N_1 = 5$	$N_2 =$	6	3/5	2/3	2/3	5/6	5/6
		7	4/7	23/35	5/7	29/35	6/7
		8	11/20	5/8	27/40	4/5	4/5
		9	5/9	3/5	31/45	7/9	4/5
		10	1/2	3/5	7/10	7/10	4/5
		15	8/15	3/5	2/3	11/15	11/15
		20	1/2	11/20	3/5	7/10	3/4
$N_1 = 6$	$N_2 =$	7	23/42	4/7	29/42	5/7	5/6
		8	1/2	7/12	2/3	3/4	3/4
		9	1/2	5/9	2/3	13/18	7/9
		10	1/2	17/30	19/30	7/10	11/15
		12	1/2	7/12	7/12	2/3	3/4
		18	4/9	5/9	11/18	2/3	13/18
		24	11/24	1/2	7/12	5/8	2/3

One-Sided Test:		$p = .90$	.95	.975	.99	.995
Two-Sided Test:		$p = .80$	.90	.95	.98	.99
$N_1 = 7$	$N_2 =$ 8	27/56	33/56	5/8	41/56	3/4
	9	31/63	5/9	40/63	5/7	47/63
	10	33/70	39/70	43/70	7/10	5/7
	14	3/7	1/2	4/7	9/14	5/7
	28	3/7	13/28	15/28	17/28	9/14
$N_1 = 8$	$N_2 =$ 9	4/9	13/24	5/8	2/3	3/4
	10	19/40	21/40	23/40	27/40	7/10
	12	11/24	1/2	7/12	5/8	2/3
	16	7/16	1/2	9/16	5/8	5/8
	32	13/32	7/16	1/2	9/16	19/32
$N_1 = 9$	$N_2 =$ 10	7/15	1/2	26/45	2/3	31/45
	12	4/9	1/2	5/9	11/18	2/3
	15	19/45	22/45	8/15	3/5	29/45
	18	7/18	4/9	1/2	5/9	11/18
	36	13/36	5/12	17/36	19/36	5/9
$N_1 = 10$	$N_2 =$ 15	2/5	7/15	1/2	17/30	19/30
	20	2/5	9/20	1/2	11/20	3/5
	40	7/20	2/5	9/20	1/2	
$N_1 = 12$	$N_2 =$ 15	23/60	9/20	1/2	11/20	7/12
	16	3/8	7/16	23/48	13/24	7/12
	18	13/36	5/12	17/36	19/36	5/9
	20	11/30	5/12	7/15	31/60	17/30
$N_1 = 15$	$N_2 =$ 20	7/20	2/5	13/30	29/60	31/60
$N_1 = 16$	$N_2 =$ 20	27/80	31/80	17/40	19/40	41/80
Large-sample approximation		$1.07\sqrt{\dfrac{m+n}{mn}}$	$1.22\sqrt{\dfrac{m+n}{mn}}$	$1.36\sqrt{\dfrac{m+n}{mn}}$	$1.52\sqrt{\dfrac{m+n}{mn}}$	$1.63\sqrt{\dfrac{m+n}{mn}}$

SOURCE. Adapted from Massey (1952).

[a] The entries in this table are selected quantities w_p of the Smirnov test statistic T for two samples, defined by Equations (6.2.1), (6.2.2), and (6.2.3). To enter the table let N_1 be the smaller sample size and let N_2 be the larger sample size. Reject H_0 at the level α if T exceeds $w_{1-\alpha}$ as given in the table. If n and m are not covered by this table, use the large sample approximation given at the end of the table.

APPENDIX E

FREQUENCY TABULATION OF THE X LENGTH OF THE PARTICLES

Frequency Tabulation

Class	Lower Limit	Upper Limit	Midpoint	Frequency	Relative Frequency	Cumulative Frequency	Cum. Rel. Frequency
at or below		.00		0	.00000	0	.000
1	.00	2.00	1.00	0	.00000	0	.000
2	2.00	4.00	3.00	174	.19142	174	.191
3	4.00	6.00	5.00	0	.00000	174	.191
4	6.00	8.00	7.00	138	.15182	312	.343
5	8.00	10.00	9.00	0	.00000	312	.343
6	10.00	12.00	11.00	133	.14631	445	.490
7	12.00	14.00	13.00	0	.00000	445	.490
8	14.00	16.00	15.00	148	.16282	593	.652
9	16.00	18.00	17.00	0	.00000	593	.652
10	18.00	20.00	19.00	82	.09021	675	.743
11	20.00	22.00	21.00	0	.00000	675	.743
12	22.00	24.00	23.00	59	.06491	734	.807
13	24.00	26.00	25.00	0	.00000	734	.807
14	26.00	28.00	27.00	37	.04070	771	.848
15	28.00	30.00	29.00	32	.03520	803	.883
16	30.00	32.00	31.00	0	.00000	803	.883
17	32.00	34.00	33.00	27	.02970	830	.913
18	34.00	36.00	35.00	0	.00000	830	.913
19	36.00	38.00	37.00	20	.02200	850	.935
20	38.00	40.00	39.00	0	.00000	850	.935
21	40.00	42.00	41.00	11	.01210	861	.947
22	42.00	44.00	43.00	0	.00000	861	.947
23	44.00	46.00	45.00	10	.01100	871	.958
24	46.00	48.00	47.00	0	.00000	871	.958
25	48.00	50.00	49.00	7	.00770	878	.966
26	50.00	52.00	51.00	0	.00000	878	.966
27	52.00	54.00	53.00	4	.00440	882	.970
28	54.00	56.00	55.00	0	.00000	882	.970
29	56.00	58.00	57.00	3	.00330	885	.974
30	58.00	60.00	59.00	3	.00330	888	.977
31	60.00	62.00	61.00	0	.00000	888	.977
32	62.00	64.00	63.00	5	.00550	893	.982
33	64.00	66.00	65.00	0	.00000	893	.982
34	66.00	68.00	67.00	5	.00550	898	.988
35	68.00	70.00	69.00	0	.00000	898	.988
36	70.00	72.00	71.00	0	.00000	898	.988
37	72.00	74.00	73.00	0	.00000	898	.988
38	74.00	76.00	75.00	1	.00110	899	.989
39	76.00	78.00	77.00	0	.00000	899	.989
40	78.00	80.00	79.00	3	.00330	902	.992
41	80.00	82.00	81.00	0	.00000	902	.992
42	82.00	84.00	83.00	0	.00000	902	.992
43	84.00	86.00	85.00	0	.00000	902	.992
44	86.00	88.00	87.00	0	.00000	902	.992
45	88.00	90.00	89.00	4	.00440	906	.997
46	90.00	92.00	91.00	0	.00000	906	.997
47	92.00	94.00	93.00	0	.00000	906	.997
48	94.00	96.00	95.00	0	.00000	906	.997
49	96.00	98.00	97.00	1	.00110	907	.998
50	98.00	100.00	99.00	0	.00000	907	.998
above	100.00			2	.00220	909	1.000

LIST OF REFERENCES

1. Edwards, T.D., Harris, K.R., Horton, K.G., Keith, M.G., Kertadidjaja, A., Lee, Y.S., Redman, D.N., "Measurements of Particles in Solid Propellant Rocket Motors", Naval Postgraduate School, Monterey, CA 93943, October 1987.

2. Redman, D.N., *Image Analysis of Solid Propellant Combustion Holograms Using an ImageAction Software Package*, Master's Thesis, Naval Postgraduate School, Monterey, CA, June 1986.

3. Edwards, T.D., *Implementation of Three Speckle Reduction Filters for Solid Propellant Combustion Holograms*, Master's Thesis, Naval Postgraduate School, Monterey, CA, June 1986.

4. Orguc, E.S., *Automatic Data Retrieval from Rocket Motor Holograms*, Master's Thesis, Naval Postgraduate School, Monterey, CA, December 1987.

5. Imaging Technology Incorporated, *The ImageAction User's Guide*, Publication 47-S00003-02, Version 2.0 May 1985.

6. Imaging Technology Incorporated, *The Itex PC Programmer's Manual*, Part Number 47-S00003-02, Version 1.1, September 1985.

7. Lee, J.S., "Speckle Suppression and Analysis for Synthetic Aperture Radar", Optical Engineering, Vol. 25, No. 5, pp. 636-643, May 1986.

8. Crimmins, T.R., "Geometric Filter for Speckle Reduction", Optical Engineering, Vol. 25, No. 5, pp. 651-654, May 1986.

9. Crimmins, T.R., "Geometric Filter for Speckle Reduction", Applied Optics, Vol. 24, No. 10, p. p. 1438 May 1985.

10. Kaeser, D.S., *Code Optimization of Speckle Reduction Algorithms for Rocket Motor Hologram Image Processing*, Master's Thesis, Naval Postgraduate School, Monterey, CA, December 1988.

11.   Conover, W.J., <u>Practical Nonparametric Statistics</u>, John Wiley & Sons, Inc., 1971.

INITIAL DISTRIBUTION LIST

		No. Copies

1. Defense Technical Information Center   2
   Cameron Station
   Alexandria, Virginia 22304-6145

2. Library, Code 0142   2
   Naval Postgraduate School
   Monterey, California 93943-5002

3. Department Chairman, Code 62   3
   Department of Electrical and
      Computer Engineering
   Naval Postgraduate School
   Monterey, California 93943-5000

4. Professor D.W. Netzer, Code 67 Nt   2
   Department of Aeronautics
   Naval Postgraduate School
   Monterey, California 93943-5000

5. Commanding Officer   1
   Air Force Rocket Propulsion Laboratory
   Edwards Air Force Base, California 93523
   Attention: Capt. M. Moser

6. Director Land Armament and Electronics   1
      Engineering and Maintenance
   National Defense Headquarters
   Ottawa, Ontario
   Canada   K1A 0K2

7. Director Personnel Education and Development   1
   National Defense Headquarters
   Ottawa, Ontario
   Canada   K1A 0K2

8. Major J.G.D. Carrier   2
   Directorate of Land Armament and Electronics
      Engineering and Maintenance   4-4
   National Defense Headquarters
   Ottawa, Ontario
   Canada   K1A 0K2