22
2ej.

Universidad Nacional Autónoma de México

Facultad de Ciencias

Evaluación de las aberraciones de un frente de onda por el método de Roddier

T E S I S

Que para obtener el grado de:

F I S I C O

P r e s e n t a :

Salvador Fierro Pacheco

México, D. F. 1993

TESIS CON
FALLA DE ORIGEN

Universidad Nacional Autónoma de México
UNAM
Biblioteca Central
Dirección General de Bibliotecas de la UNAM

UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Contenido

Pags.

Capítulo UNO. Introducción.

1.1 Antecedentes. 1

Capítulo DOS. El Método de Roddier.

2.1 Introducción. 3
2.2 El Método de Roddier. 3
2.3 Ecuación de transporte de irradiancia (E.T.I.). 4
2.4 Derivación de la fórmula de Roddier a partir de E.T.I. 5
2.5 Expresión base de la tesis. 8
2.6 Funcionamiento del método en la tesis. 9

Capítulo TRES. Instrumentación del Método.

3.1 Sistema Optico. 11
3.2 Adquisición de Datos. 11
3.3 Sistema de Procesamiento. 13
3.4 Despliegue. 14
3.5 Muestreo. 16
3.6 Procesamiento de Datos. 18

Capítulo CUATRO. Verificación Experimental.

4.1 Introducción. 22
4.2 Test interferométrico. 22
4.3 El método de Roddier. 23
4.4 Resultados. 24
4.4 Conclusiones. 24

Apéndice A. Polinomios de Zernike.

A.I Introducción. 29
A.II Polinomios de Zernike. 29
A.III Derivada de Zernike. 30
A.IV Cálculo de $\nabla^2\mathcal{W}$ y $\frac{\partial \mathcal{W}}{\partial r}$ 31

Apéndice B. Métodos Numéricos.

B.I Método de mínimos cuadrados general. 33
B.II Descomposición LU 36

Apéndice C. Interfaz gráfica de usuario.

C.I Programa principal. (RODDIER.PAS). 40
C.II Fondo módulo principal (Unidad Bk). 43
C.III Ventanas de ayuda e instrucciones (Unidad AYUDA). 45
C.IV Utilerías varias. (Unidad MENUDO). 46
C.V Manejo de la tarjeta DT2851 (Unidad IRIS). 48
C.VI Muestreo de la imagen (Unidad SCAN2). 50
C.VII Diferencia de intensidad de un par de imgenes (Unidad MRDD3). 56
C.VIII Despliegue de la imagen (Unidad DISP). 58
C.IX Cálculo del tiempo de procesamiento (Unidad ZEIT) 60
C.X Encabezado común de configuración (Unidad HEAD). 61

Apéndice D. Divisor de cubo.

D.I Cálculo de aberraciones. 62

Bibliografía

CAPITULO UNO

Introducción.

1.1 Antecedentes.

Caracterizar un frente de onda por medio de pruebas interferométricas, no presenta mayores dificultades, si se aplica a sistemas con un diámetro menor a 40 cm. Cuando el sistema es mayor, tómese como ejemplo un telescopio de 1m. de diámetro, surgen varios inconvenientes en la aplicación de técnicas interferométricas

- La necesidad de un espejo plano (a $\lambda/10$ al menos) con un diámetro mayor o igual a 1m. para regresar el haz colimado al interferómetro de prueba. *
- La utilización de un laser estabilizado, ésto es con una longitud de coherencia mayor a 30m. (en este caso para un primario de 1m., la distancia focal es de alrededor 15m.).
- Hay una fuerte dependencia con la turbulencia atmosférica y las vibraciones, enmascarando las aberraciones necesarias en la evaluación.

Considerando la problemática anterior, se buscó un método de evaluación de aberraciones aplicable a diferentes sistemas, sin importar el tamaño de la pupila. Se empezó a sospechar que las imágenes desenfocadas de estrellas podrían servir en la medición de aberraciones. Ruiz y Cuevas[1] describen el efecto de coma en imágenes fuera de foco, proponiendo lo anterior como test para alinear la óptica del telescopio de 2.1 m. de S.P.M. Beckers[2] describe imágenes desenfocadas registradas en el MMT, interpretando las zonas observadas como errores de curvatura local en el frente de onda.

F. Roddier[3] propuso finalmente un método alternativo para evaluar el frente de onda basado en imágenes desenfocadas simétricas. Este método tiene muchas ventajas, así como una mayor sencillez en su montaje, comparado con la prueba interferométrica. La principal de ellas es la supresión del espejo de regreso del haz y la independencia de las condiciones de turbulencia atmosférica. Además no se necesita ningún laser, ya que se puede utilizar una fuente puntual de luz blanca.

En esta tesis se instrumentó un método basado en la prueba de Roddier para la evaluación de un frente de onda con las siguientes características.

- Aplicable al laboratorio en ambientes de luz reducida.
- Gran facilidad de uso basado en una interfaz gráfica de usuario, además de incorporar el ratón en el uso de menus de selección.
- Almacenamiento de datos en el disco duro de la PC.
- Adquisición de imágenes mediante el control de una tarjeta digitalizadora en la captura de imágenes procedentes de una cámara CCD o videocasetera.

---

* Véase D. Malacara, "Optical Shop Testing", p.750 2nd. edition, Wiley Publ. (1992).

- Salida de resultados en formato texto funcionando como bitácora.

Este trabajo se limitó a evaluar sistemas en laboratorio, con la esperanza de aplicarlo en el futuro para la prueba de los telescopios del IAUNAM en S.P.M. y Tonantzintla. La idea es establecer un método análogo al de Roddier basándose en polinomios de Zernike y calibrar el resultado con una prueba interferométrica. A continuación se describe el contenido de los capítulos para lograr el objetivo anterior.

En el capítulo dos se describe el método de Roddier, su derivación a partir de la ecuación de transporte de irradiancia y la formulación final como expresión base de la tesis. Para reconstruir el frente de onda se requiere resolver la ecuación de Poisson con condición de frontera de Neumann. En este trabajo se desarrolla un método alternativo utilizando un sistema de ecuaciones simultáneas con términos de Zernike

La instrumentación del método es objeto del capítulo tres. Ahí se explica el sistema óptico y el sistema desarrollado para la medición. Además se detallan cada una de las etapas implicadas en el procesamiento de las imágenes desenfocadas y la obtención de la aberraciones.

En el capítulo cuatro, se aplica el método a una superficie en el laboratorio, calculándose las aberraciones, y estableciendo una comparación con los resultados de la prueba interferométrica a la misma superficie. Finalmente se concluye con los alcances del método.

El apéndice A describe la utilidad de los polinomios de Zernike y como intervienen en la expresión base de la tesis. En el apéndice B se explica la utilización de métodos numéricos para el cálculo de la aberraciones. El apéndice C contiene las rutinas y programas utilizados por el sistema en la evaluación de los términos de Zernike. Por último la justificación del uso de un divisor de cubo en el sistema óptico es tema del apéndice D.

CAPITULO DOS

El Método de Roddier

2.1 Introducción.

La evaluación cualitativa de aberraciones de un telescopio mediante el uso de imágenes desenfocadas es una práctica conocida en óptica. Sin embargo hay pocos intentos en extraer información cuantitativa de tales observaciones. Los trabajos de Behr[4] y Ruiz y Cuevas[1] son ejemplos interesantes de ello. Beckers *et al.* describen imágenes desenfocadas procedentes del MMT proporcionando una interpretación de las zonas observadas como errores de curvatura local en el frente de onda. Más recientemente F. Roddier[3] ha propuesto un método general de reconstrucción de un frente de onda basada en imágenes fuera de foco.

La presente tesis tiene como objeto evaluar las aberraciones de un sistema óptico o una superficie que se esta puliendo, basándose en una expresión análoga a la de Francois Roddier y sistematizar un procedimiento de medida. En este capítulo se describe el método de Roddier de manera concisa. Se demuestra la formulación de Roddier a partir de la ecuación de transporte de irradiancia. Esta ecuación da una relación entre las diferencias de intensidad de dos imágenes desenfocadas y la aberración del frente de onda. Estas aberraciones son descritas como un polinomio de Zernike, véase apéndice A. Existen dos caminos alternativos para la resolución de la ecuación de Roddier, el primero es resolver la ecuación de Poisson con una condición de frontera de Neumann, la segunda y seguida en la presente tesis es usar polinomios de Zernike, planteando un sistema de ecuaciones lineales simultáneas que se resuelve por un método numérico.

2.2 El método de Roddier.

Básicamente el método consiste en la medición de iluminación de $I_1(\vec{r})$ en el plano P_1, a una distancia ℓ antes del plano focal F y la medición de $I_2(-\vec{r})$ simétricamente colocada en P_2 a una distancia ℓ después de F como en la figura 2.1.

Los errores de curvatura locales en el frente de onda $\mathcal{W}$ hacen converger a la luz más cerca en un plano que en otro, presentándose un exceso de iluminación en un plano y una carencia de iluminación en el otro, siendo éste par de imágenes complementarias. La diferencia de iluminación proporciona una medida de curvatura local del frente de onda.

De acuerdo a la óptica geométrica y suponiendo[5] $\ell \ll f$

$$\frac{\Delta I}{I}(\vec{\rho}) = 2\frac{\lambda f^2}{\ell R^2}\nabla^2\mathcal{W}(\vec{\rho}) \tag{2.1}$$

donde $\Delta I = I_1(\vec{r}) - I_2(-\vec{r})$, $I = \frac{1}{2}[I_1(\vec{r}) + I_2(-\vec{r})]$ es la iluminación promedio, $\vec{\rho}$ es el vector de posición en la pupila del telescopio expresada en unidades de radio de la pupila, R es el radio de la pupila y $\mathcal{W}$ es el frente de onda en unidades de longitud de onda λ.

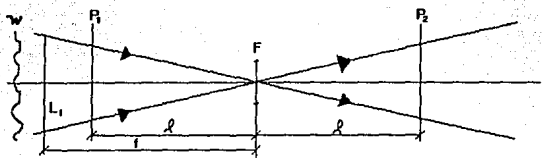

Figura 2.1. La diferencia de iluminación entre los planos P_1 y P_2 es una medida de la distribución de curvatura local en el frente de onda entrante. También mide tilt radial en la orilla del frente de onda.

El resultado es independiente de la heterogeneidad de la distribución de luz, en el plano de la pupila, debido a que no es uniforme la distribución en las dos imágenes desenfocadas y se cancela cuando se hace la diferencia. F. Roddier sugiere como método de reconstrucción de W la resolución de la ecuación diferencial 2.1 (ecuación de Poisson) con la condición de frontera de Neumann[5]. Por otro lado, la resolución de la ecuación diferencial no es la única alternativa para conocer $\mathcal{W}$. Aquí se propone un camino alternativo descrito en las secciones siguientes.

2.3 Ecuación de transporte de irradiancia.

Ichikawa *et al.* han demostrado experimentalmente una ecuación que relaciona la irradiancia local y las deformaciones de fase de un frente de onda[6], el frente se propaga en la dirección z como se muestra en la figura 2.2. Esta ecuación esta dada por

$$\frac{\partial I}{\partial z} = -(\nabla I \cdot \nabla \mathcal{W} + I \nabla^2 \mathcal{W}) \tag{2.2}$$

donde $I(x, y, z)$ es la distribución de iluminación a lo largo del haz y $\mathcal{W}(x, y, z)$ describe el frente de onda a una distancia Δz medida a partir del origen. Esta ecuación se le llama *ecuación de transporte de irradiancia.*

La derivada $\frac{\partial I}{\partial z}$ se mide a partir de la diferencia de iluminación. La primera medición se hace en el plano $z = 0$ y la segunda en $z = \Delta z$. Asimismo $\frac{\partial I}{\partial z}$ es el resultado también de la contribución de inclinación $\nabla \mathcal{W}$ y el término $\nabla^2 \mathcal{W}$ en la región paraxial.

Es importante conocer en la utilización de este principio el significado físico de cada elemento de la ecuación de transporte de irradiancia. El haz paraxial en dirección de z se puede describir con una amplitud compleja como:

$$\sqrt{I(x, y, z)} \exp\ (ik\mathcal{W}(x, y, z)) \tag{2.3}$$

donde $I(x, y, z)$ es la irradiancia del haz, $\mathcal{W}$ la fase en unidades de λ y $k = 2\pi/\lambda$ el número de onda.

La expresión 2.2 se puede escribir como

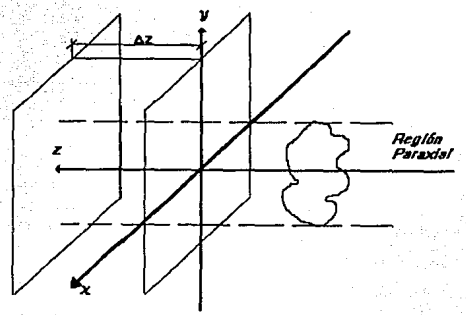

Figura 2.2. Ecuación de transporte de Irradiancia.

$$\nabla_T I \cdot \nabla_T \mathcal{W} + I \nabla^2 \mathcal{W} + \frac{\partial I}{\partial z} = 0 \tag{2.4}$$

siendo $\nabla_T \equiv \frac{\partial}{\partial x} + \frac{\partial}{\partial y}$ el operador gradiente en el plano XY normal a la dirección de propagación del haz.

El primer término $\nabla_T I \cdot \nabla_T \mathcal{W}$ representa la variación de la intensidad causada por el corrimiento transversal de un haz heterogéneo. Este corrimiento se debe a las inclinaciones o cambios de pendiente locales del frente de onda, cuya normal esta dada por ∇_T. A este se le conoce como *término de prisma*. Es válido interpretarlo como el efecto causado por un prisma sobre un haz de luz que lo atraviesa. El segundo elemento $I\nabla_T^2\mathcal{W}$ o *término de lente* es la variación en la fase provocada por una lente[7]. Entonces $\frac{\partial I}{\partial z}$ es análogo al efecto conjunto de un prisma y una lente sobre un haz cuya propagación es sobre el eje z.

La ecuación 2.4 es posible reescribirla como

$$\nabla_T \cdot (I \nabla_T \mathcal{W}) = -\frac{\partial I}{\partial z} \tag{2.5}$$

La ecuación de transporte de irradiancia en la forma anterior representa una ley de conservación de energía, análoga a las leyes de conservación de masa o carga, como por ejemplo $\mathrm{div}(\rho v) = -\frac{\partial \rho}{\partial t}$, con ρ, v densidad de carga y velocidad de flujo respectivamente.

2.4 Derivación de la fórmula de Roddier a partir de la ec. de transporte de irradiancia.

A partir de la relación (2.5) se puede demostrar la formula (2.1) dada por Roddier. Iniciando con la hipótesis de una iluminación uniforme I_0 dentro de la imagen desenfocada y un valor nulo fuera de ella, se puede definir la siguiente función ∇I

$$\nabla I = \begin{cases} -I_0 \vec{n} \delta_c & \text{en el borde de la imagen desenfocada} \\ 0 & \text{fuera de ella} \end{cases} \tag{2.6}$$

δ_c es la distribución lineal de Dirac válida solo en la orilla de la pupila, $\vec{n}$ es el vector unitario perpendicular al borde apuntando hacia el exterior.

Sea P la función de transmisión de la pupila como sigue

$$P = \begin{cases} 1, & \text{dentro de la pupila;} \\ 0, & \text{fuera de la pupila.} \end{cases}$$

Incorporando (2.6) en (2.2) se establece

$$\frac{\partial I}{\partial z} = I_0 \delta_c \vec{n} \cdot \nabla \mathcal{W} - I^2 \nabla^2 \mathcal{W} \tag{2.7}$$

si $\frac{\partial \mathcal{W}}{\partial n} = \vec{n} \cdot \nabla \mathcal{W}$ y $I = I_0 P$ entonces

$$\frac{\partial I}{\partial z} = I_0 \left[\frac{\partial \mathcal{W}}{\partial n} \delta_c - P \nabla^2 \mathcal{W} \right] \tag{2.8}$$

Dentro de (2.8), $\frac{\partial \mathcal{W}}{\partial n}$ representa la derivada direccional perpendicular a la orilla de la pupila desenfocada.

Las mediciones de las imágenes desenfocadas I_1, I_2 son como siguen

$$\begin{aligned} I_1 &= I_0 + \frac{\partial I}{\partial z} \Delta_z \\ I_2 &= I_0 - \frac{\partial I}{\partial z} \Delta_z \end{aligned} \tag{2.9}$$

Si se define la señal del sensor del frente de onda S como

$$S = \frac{I_1 - I_2}{I_1 + I_2} = \frac{1}{I_0} \frac{\partial I}{\partial z} \Delta_z \tag{2.10}$$

De (2.8) y (2.10) se obtiene

$$S = \left(\frac{\partial \mathcal{W}}{\partial n} \delta_c - P \nabla^2 \mathcal{W} \right) \Delta_z \tag{2.11}$$

$\frac{\partial \mathcal{W}}{\partial n}$ apunta *hacia afuera* del borde de radio R_1, ver figura 2.3.

Por geometría es posible obtener

$$S = \frac{I_1(\vec{r}) - I_2(-\vec{r})}{I_1(\vec{r}) + I_2(-\vec{r})} \tag{2.12}$$

el valor $-\vec{r}$ es debido al cambio de signo producido por la rotación de 180° entre el plano P_1 y P_2, véase la figura 2.3. Conjuntando (2.10) y (2.12)

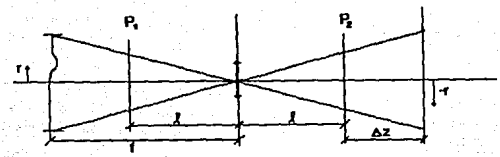

Figura 2.3. Representación geométrica del método de Roddier.

$$\frac{I_1(\vec{r}) - I_2(-\vec{r})}{I_1(\vec{r}) + I_2(-\vec{r})} = \frac{1}{I_0}\frac{\partial I}{\partial z}\Delta_z$$

P_1 tiene un radio R_1 dado por

$$\frac{R}{f} = \frac{R_1}{\ell} \Rightarrow R_1 = \frac{\ell}{f}R$$

con un radio R correspondiente al tamaño de la pupila real. El factor de escalamiento anterior es necesario en (2.11). Además en la figura 2.3 ocurre que $\Delta z = f - \ell$, con lo cual se tiene

$$S = \frac{(f-\ell)}{\ell} f \left[\frac{\partial \mathcal{W}}{\partial n}\delta_c - P\nabla^2\mathcal{W}\right] \tag{2.13}$$

Tomando los valores adecuados de las definiciones de δ_c y P sobre la pupila:

$$S = -\left(\frac{f-\ell}{\ell}\right) f \left[\nabla^2\mathcal{W} - \frac{\partial I}{\partial z}\bigg|_{r=R}\right] \tag{2.14}$$

arreglando las constantes y normalizando el radio con $\rho = \frac{r}{R}$ además de expresar $\mathcal{W}$ en unidades de λ, se llega a:

$$S = -\frac{(f-\ell)f\lambda}{\ell R^2}\left\{\nabla^2\mathcal{W}(\rho,\theta) - \frac{\partial \mathcal{W}(\rho,\theta)}{\partial \rho}\bigg|_{\rho=1}\right\} \tag{2.15}$$

en nuestro caso se cumple $f \gg \ell$ y la expresión anterior se transforma en

$$S = \frac{\lambda f^2}{\ell R^2}\left\{\nabla^2 \mathcal{W}(\rho,\theta) - \frac{\partial \mathcal{W}(\rho,\theta)}{\partial \rho}\bigg|_{\rho=i}\right\} \tag{2.16}$$

esta expresión coincide con la ecuación (5) de Roddier[3]. Por tanto la expresión base del *método de Roddier* es

$$\frac{I_1(\bar{\rho}) - I_2(-\bar{\rho})}{\frac{1}{2}\left(I_1(\bar{\rho}) + I_2(-\bar{\rho})\right)} = \frac{2\lambda f^2}{\ell R^2}\left\{\nabla^2 \mathcal{W}(\rho,\theta) - \frac{\partial \mathcal{W}}{\partial \rho}\bigg|_{\rho=1}\right\} \tag{2.18}$$

siendo (f, R) parámetros conocidos del sistema óptico y ℓ la posición de las imágenes desenfocadas con respecto al foco, λ es la longitud de onda usada.

2.5 *Expresión base de la tesis.*

La manera natural de obtener $\mathcal{W}$ es la sugerida por F. Roddier, consistente en resolver la ecuación de Poisson (2.1) con la condición de Neumann. Un segundo camino es expresar al frente de onda $\mathcal{W}$ en función de los polinomios de Zernike y calcular la expresión (2.18) aplicando ∇^2 y $\frac{\partial}{\partial \rho}$ a $\mathcal{W}$. El conjunto de polinomios de Zernike es muy extenso, basta sólo considerar ocho términos $(Z_1, \cdots Z_8)$ para expresar con un error pequeño (0.62%) el frente de onda[8]. La aproximación con Zernike es así hasta tercer orden. La asociación Aberración clásica-Zernike para aberraciones se muestra en la tabla 2.1.

El cálculo basado en sólo esas aberraciones proporciona una buena caracterización del frente de onda. El frente de onda queda como $\mathcal{W}(r,\theta) = \sum_{i=1}^{8} c_i Z_i(\rho,\theta)$ con Z_i como sigue en la tabla 2.1.

Zernike	*Aberración*
$Z_1 = 2c_1\rho\cos\theta$	tilt sobre x
$Z_2 = 2c_2\rho\mathrm{sen}\theta$	tilt sobre y
$Z_3 = \sqrt{3}c_3(2\rho^2 - 1)$	desenfoque
$Z_4 = \sqrt{6}c_4\rho^2\cos 2\theta$	astigmatismo sobre x
$Z_5 = \sqrt{6}c_5\rho^2\mathrm{sen}2\theta$	astigmatismo sobre y
$Z_6 = \sqrt{8}c_6(3\rho^2 - 2)\rho\cos\theta$	coma sobre x
$Z_7 = \sqrt{8}c_7(3\rho^2 - 2\rho\mathrm{sen}\theta$	coma sobre y
$Z_8 = \sqrt{5}c_8(6\rho^4 - 6\rho^2 + 1)$	aberración esférica

Tabla 2.1. Asociación Aberración clásica-Zernike (tercer orden).

La parte de curvatura se calcula aplicando el operador laplaciano en coordenadas polares debido a la simetría del problema. El cálculo de ∇^2 se encuentra en A.IV cuyo resultado esta en la tabla 2.2

$$\nabla^2 Z_1 = \nabla^2 Z_2 = \nabla^2 Z_4 = \nabla^2 Z_5 = 0$$
$$\nabla^2 Z_3 = 8\sqrt{3}c_3$$
$$\nabla^2 Z_6 = 24\sqrt{8}c_6\rho\cos\theta$$
$$\nabla^2 Z_7 = 24\sqrt{8}c_7\rho\mathrm{sen}\theta$$
$$\nabla^2 Z_8 = 24\sqrt{5}c_8(4\rho^2 - 1)$$

Tabla 2.2. Laplaciano de $\nabla^2\mathcal{W}$.

Es notoria la anulación de los términos de tilt y astigmatismo, además de la imposibilidad de determinarlos con sólo la parte de curvatura. La contribución de la frontera se obtiene calculando $\frac{\partial\mathcal{W}}{\partial n}\Big|_{\rho=1}$ cuyo resultado se ve en la siguiente tabla.

$\frac{\partial}{\partial n}Z_1\Big\vert_{\rho=1} = 2c_1\cos\theta$	$\frac{\partial}{\partial n}Z_5\Big\vert_{\rho=1} = 2\sqrt{6}c_5\mathrm{sen}2\theta$
$\frac{\partial}{\partial n}Z_2\Big\vert_{\rho=1} = 2c_2\mathrm{sen}\theta$	$\frac{\partial}{\partial n}Z_6\Big\vert_{\rho=1} = 7\sqrt{8}c_6\cos\theta$
$\frac{\partial}{\partial n}Z_3\Big\vert_{\rho=1} = 4\sqrt{3}c_3$	$\frac{\partial}{\partial n}Z_7\Big\vert_{\rho=1} = 7\sqrt{8}c_7\mathrm{sen}\theta$
$\frac{\partial}{\partial n}Z_4\Big\vert_{\rho=1} = 2\sqrt{6}c_4\cos 2\theta$	$\frac{\partial}{\partial n}Z_8\Big\vert_{\rho=1} = 12\sqrt{5}c_8$

Tabla 2.3. Cálculo de $\frac{\partial\mathcal{W}}{\partial n}$.

Conjuntando los resultados de las tablas 2.2 y 2.3, la expresión base del método de Roddier es la ecuación 2.19.

$$S = \frac{I_1(\vec{r}) - I_2(-\vec{r})}{\frac{1}{2}\left\{I_1(\vec{r}) + I_2(-\vec{r})\right\}} = 2\frac{\lambda f^2}{lR^2}\Big\{-2c_1\cos\theta + 2c_2\mathrm{sen}\theta + 4\sqrt{3}c_3 +$$
$$-2\sqrt{6}c_4\cos 2\theta + -2\sqrt{6}c_5\mathrm{sen}2\theta + 24\sqrt{8}c_6\rho\cos\theta - 7\sqrt{8}c_6\cos\theta +$$
$$+24\sqrt{8}c_7\rho\mathrm{sen}\theta - 7\sqrt{8}c_7\mathrm{sen}\theta + 24\sqrt{5}c_8(4\rho^2 - 1) - 12\sqrt{5}c_8\Big\} \quad (2.19)$$

2.6 Funcionamiento del método en la tesis.

Al aplicar un ajuste por mínimos cuadrados a la ecuación (2.19) se obtiene el valor de los coeficientes $\{c_1 \ldots c_8\}$. Cada coeficiente expresa el peso de cada aberración en el frente

de onda $\mathcal{W}$. Los datos para alimentar dicho ajuste es el conjunto $(r, \theta, I(r, \theta))$ de cada par de imágenes.

La instrumentación del método con el objetivo de conocer el conjunto de c_i's es tema del capítulo TRES. El análisis y discusión de dichos resultados tiene lugar en el capítulo CUATRO.

CAPITULO TRES

Instrumentación del Método

3.1 Sistema Optico.

Como se mencionó en el capítulo DOS, el método de Roddier consiste en procesar imágenes desenfocadas intra y extrafocales simétricas, al plano focal del sistema óptico que se desea examinar. En esta aplicación del método, el sistema óptico puede ser un telescopio astronómico o una superficie óptica en proceso de pulido en el taller. En el caso del telescopio astronómico, se enfoca sobre una cámara CCD cerca del plano focal la imagen desenfocada una estrella brillante. La cámara CCD debe montarse sobre un sistema de desplazamiento a lo largo del eje óptico. La aplicación de este método sobre una superficie óptica se explica en detalle en el capítulo CUATRO.

En este capítulo se muestra la instrumentación del método de Roddier. Se utilizó una computadora compatible PC Acer 386-33 MHz con 6 MbRAM, disco duro de 100 Mb, tarjeta de video Genoa Super VGA, monitor Sony Multisync. La computadora esta equipada con una tarjeta digitalizadora de imagen DATA TRANSLATION DT2851. También se hace uso de una cámara CCD y una video casetera JVC Super VHS HiFi. Se desarrolló un conjunto de programas en forma de módulos en lenguaje PASCAL que incluye interfaces de video y ratón. La figura 3.1 muestra como se conjuntan los elementos anteriores para la instrumentación del método.

Un par de imágenes intra y extrafocales (telescopio astronómico o de una superficie bajo prueba en el taller), son adquiridas por una cámara CCD, éstas son almacenadas en video o enviadas directamente a la computadora PC. Las imágenes son capturadas por la tarjeta digitalizadora, la cual es controlada por el *sistema modular de programación*. La información de cada imagen se procesa por medio del conjunto modular, proporcionando resultados en forma de archivo de texto.

La adquisición de imágenes es el tema de la siguiente sección, la forma de como se introducen los parámetros λ, f, R, ℓ descritos en el capítulo DOS, se puede ver en la sección 3.3. La visualización del contenido de cada imagen es parte del inciso 3.4; así como el muestreo de la misma en el 3.5. Finalmente la descripción del procesamiento de datos y generación de resultados se trata en 3.6.

3.2 Adquisición de datos.

Esta etapa tiene la finalidad de capturar las imágenes intra y extrafocales de la siguiente manera: se enfoca la imagen sobre el plano de la cámara CCD, la señal se hace llegar a la tarjeta digitalizadora DT2851[9]. La tarjeta DT2851 controlada por programación, convierte la señal analógica a un formato accesible de video para la computadora. Un aspecto interesante es la posibilidad de grabar en video, la sesión entera de laboratorio o

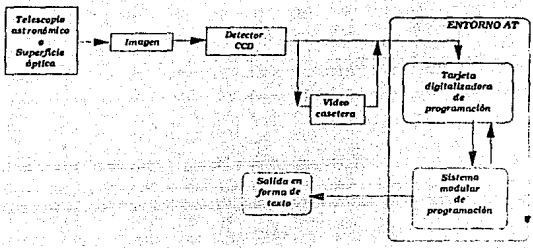

Figura 3.1. Instrumentación del método.

imágenes procedentes del telescopio por medio de una videocasetera, complementando el proceso en cualquier instante.

Las características contenidas en el módulo de captura son:

- Acceso a menús por medio del ratón.
- Minimizar el uso de teclado.
- Posibilidad de capturar imágenes en una sesión en ausencia completa de luz.
- Actualización de la imagen recibida en el monitor en un corto intervalo de tiempo.
- Observar la imagen tal y como se guarda en un archivo.
- Almacenamiento de las imágenes en el disco duro de la PC.
- Asignación de un identificador a la imagen por medio de una ventana de captura.

Todo ello dentro de una interfaz gráfica de usuario (GUI). Este módulo se activa por medio de un click sobre la opción de *captura* del menú principal como en la figura 3.2. Activada esta opción aparece una ventana de captura, con el fin de asignar un nombre a la imagen, sugiriendo simultáneamente un nombre para la serie. Como convención se utilizan un par de letras y el valor de ℓ para formar el identificador, por ejemplo para una imagen usando una lámpara de mercurio y $\ell = 135$ mm el nombre es HG135. La extensión es asignada de acuerdo a la opción de un segundo menú especificando el tipo de imagen (intra o extrafocal), completando el identificador con las extensiones .IMI o .IME respectivamente. Durante esta etapa la imagen del monitor se actualiza de manera periódica. Para capturar la imagen basta con sólo oprimir el ratón, almacenando el resultado en disco regresando al menú principal. La programación de la tarjeta DT2851 esta contenida en el apéndice C.V en el módulo de captura.

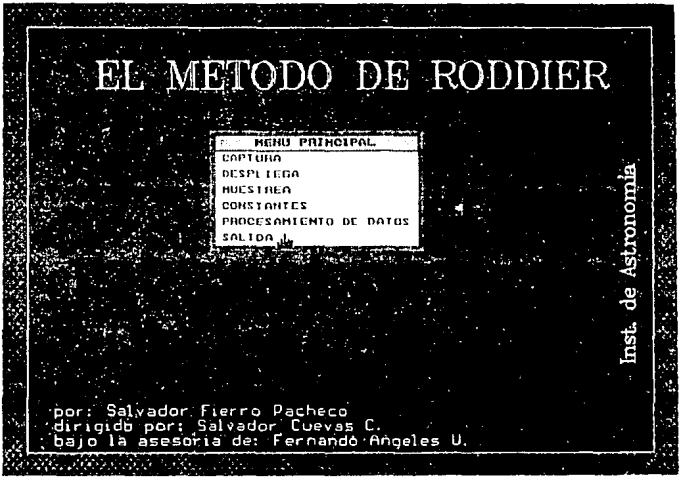

Figura 3.2. Menú principal.

3.3 Sistema de procesamiento.

El sistema de procesamiento de datos consta de cinco módulos.

- Captura de imágenes
- Constantes
- Despliegue
- Muestreo
- Procesamiento de datos

Las características de los módulos restantes son similares a las pedidas al módulo de captura. El módulo de constantes tiene la función de guardar el valor de las constantes (λ, ℓ, D, f), las cuales van asociadas a cada par de imágenes desenfocadas.

Cada constante esta expresada en mm, véase la figura 3.3. Se almacena el valor en un archivo en disco de acuerdo a la siguiente expresión

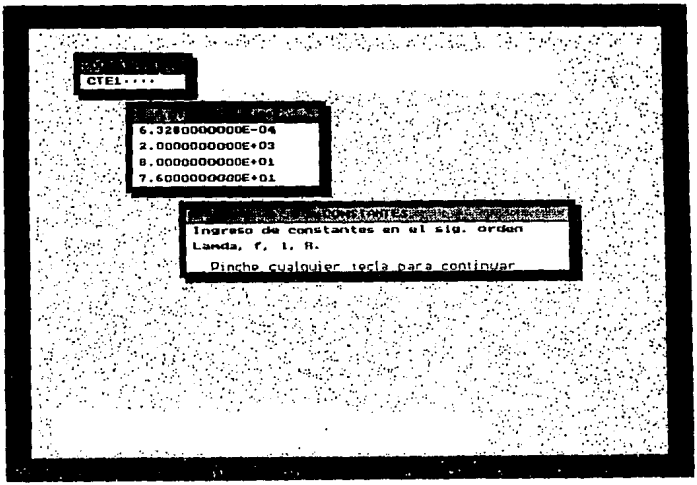

Figura 3.3. Captura de constantes.

$$2\frac{\lambda f^2}{\ell\left(\frac{D}{2}\right)^2} \tag{3.1}$$

El identificador de este archivo empieza con el nombre de la serie de imágenes y extensión CTE, ejemplo CTE1.CTE, véase la figura 3.3.

3.4 Despliegue.

La utilidad de este módulo es poder verificar el contenido de las imágenes capturadas, con el fin de seleccionarlas para el período de muestreo, véase la figura 3.4. Este módulo también permite revisar imágenes cuya iluminación no haya saturado el detector CCD.

Para ejecutar este módulo se inicia primero un mapeo de la imagen contenida en disco sobre la tarjeta de video (Genoa Super VGA). Surge una ventana mencionando la aplicación de la técnica de falso color en el despliegue de la imagen, en seguida se procede al despliegue

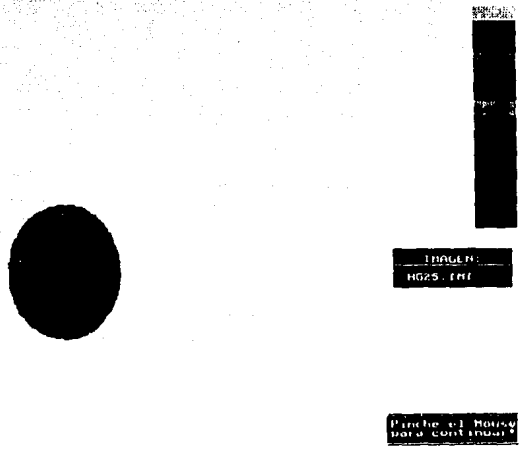

Figura 3.4. Despliegue de la imagen.

de ésta, veáse figura 3.5. Se vacía la imagen en el monitor, en el estado en que se almacenó en disco. Terminado el mapeo se aplica la técnica de falso color a la imagen (16 colores arbitrarios). Esto se debe a que la tarjeta digitalizadora es de 8 bits y sólo se tienen 256 intensidades. En una paleta de 16 valores se asigna un color a cada intervalo de 16 valores, por ejemplo el rojo a los 16 primeros y el blanco a los 16 últimos. También es posible tener 16 tonos de verde, rojo, azul, así como tonos de gris.

La técnica de falso color tiene la finalidad de reconocer de manera fácil la distribución de iluminación por zonas en la imagen, cualidad importante en el método de Roddier, debido a la obtención de las aberraciones a partir de la diferencia de iluminación entre una imagen intra y extrafocal. Esta situación se refleja en que ambas imágenes sean complementarias, las figuras 3.6a y 3.6b muestran ejemplos de imágenes intra y extrafocales.

Para explicar de manera simple la complementaridad de dos imágenes (mencionado en el capítulo DOS), úsese el siguiente ejemplo: divídase la primera imagen en cuatro cuadrantes como en un plano cartesiano. Sea la iluminación uniforme en el segundo, tercero y cuarto

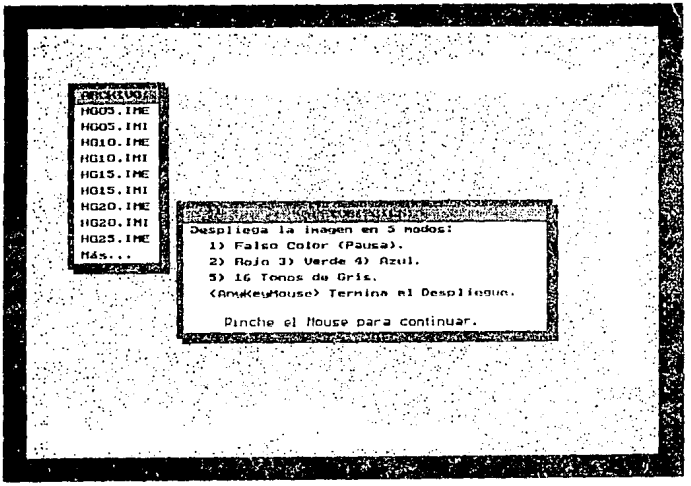

Figura 3.5. Técnica de falso color.

cuadrantes. En el primero existe un círculo con una iluminación creciente en la dirección radial, alcanzando el máximo valor en el centro del círculo. La imagen complementaria tiene una iluminación uniforme en los cuadrantes primero, segundo y cuarto, mientras en el tercero debe observarse una mancha circular cuya iluminación decrece conforme el radio, con un mínimo de intensidad en la parte central.

3.5 Muestreo.

El propósito de esta etapa es la determinación del área de muestreo de cada imagen desenfocada. Sobre esa región se muestrea un número finito de pixeles con (r, θ) específicos. Para ejecutar esta tarea se despliega la imagen sobre la tarjeta de video. Al término del mapeo se notifica al usuario mediante una señal sonora, el inicio del muestreo apareciendo simultáneamente el cursor y un polígono de radio variable. El tamaño del polígono varía conforme al movimiento del ratón, fijándose la longitud del radio del polígono (proyección sobre el eje X) mediante un *click* al ratón, ver figura 3.7

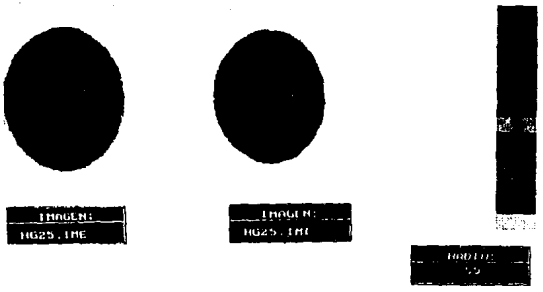

Figura 3.6. Imagen extrafocal e imagen intrafocal complementarias.

En ese momento aparece un menú cuyas opciones son repetir, abortar o salvar el resultado del muestreo. Determinada el área y posición del polígono se procede a vaciar la imagen en el monitor con una paleta de 256 colores en modo Super VGA(640 × 480 × 256), para tomar la lectura del conjunto de 60 pixeles.

Debe considerarse en el instante de muestreo, la deformación en el aspecto de un objeto sobre el monitor, cuya razón de aspecto es 1:1.257 sobre la dirección Y. Esto se observa en la transformación del polígono en una elipse final, al aparecer el menú de muestreo, véase figura 3.8.

El efecto de complementaridad en las imágenes es considerado mediante una rotación de 180° como ángulo inicial de muestreo en la intrafocal. Lo anterior es con la intención de hacer coincidir los puntos de muestreo para la diferencia de iluminación.

Definido el ángulo inicial para cada imagen, la lectura del valor del pixel se realiza sobre cinco círculos concéntricos con un radio normalizado, *i.e.* el primer círculo tiene un radio igual a 1/5, el segundo de 2/5 y así sucesivamente. En cada trayectoria circular

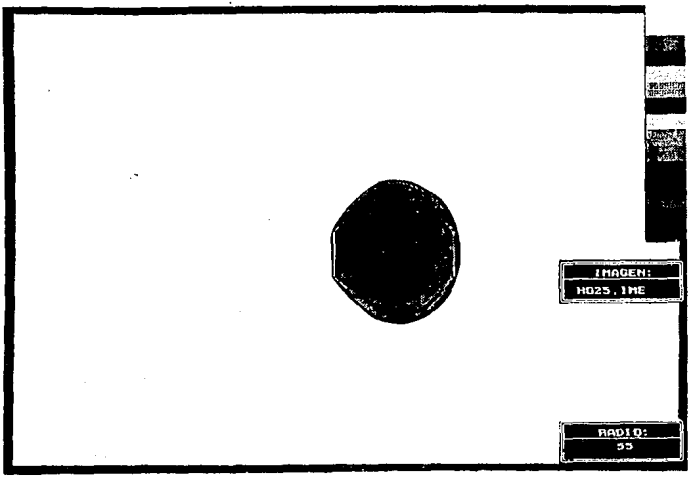

Figura 3.7. Polígono de muestreo.

se lee un pixel cada treinta grados, generando doce puntos por círculo y un total de sesenta puntos por imagen. En la medición del valor de cada pixel se promedian los ocho pixeles adyacentes junto con el pixel central a fin de suavizar la lectura. Determinada cada posición, se almacena el valor del pixel, posición y ángulo en un archivo en el disco duro.

3.6 Procesamiento de datos.

Para la generación de los coeficientes de Zernike es necesaria la información contenida en archivos producidos por los módulos de constantes y muestreo. El contenido de dichos archivos es $\{r, \theta, I(r,\theta)\}$ (sección 3.5) y las constantes $\{\lambda, \ell, f, D\}$ (sección 3.3). El polinomio considerado es

$$\frac{\Delta I}{I} = 2\frac{\lambda f^2}{\ell(\frac{D}{2})^2}\left\{\nabla^2 \mathcal{W}(\vec{\rho}) + \frac{\partial \mathcal{W}}{\partial n}\Big|_{\rho=1}\right\} \tag{3.2}$$

con $\Delta I = I_1(\vec{r}) - I_2(-\vec{r})$ e $I = \frac{1}{2}(I_1 + I_2)$.

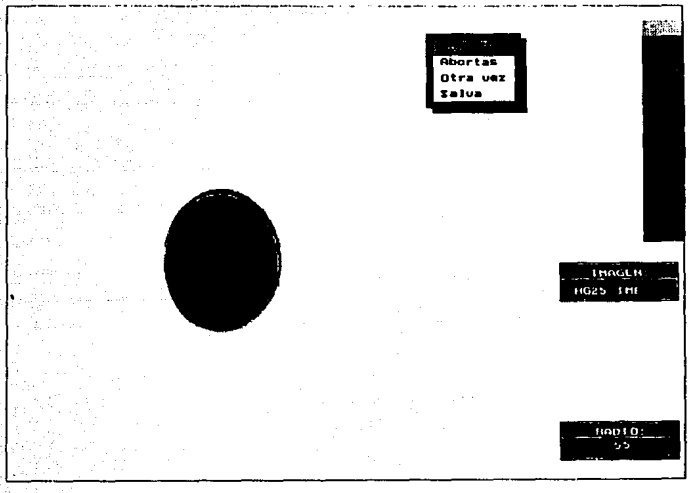

Figura 3.8. Opciones disponibles en el método de muestreo.

Los términos de $\nabla^2 W$ y $\frac{\partial W}{\partial n}$ se especifican con detalle en el capítulo DOS. Los valores de ΔI, I se calculan a partir de los datos generados en el módulo de muestreo (apéndice C.VI)

La información de entrada de esta última etapa, la constituye el conjunto de datos $\left\{\left(\frac{\Delta I_i}{I_i}, r_i, \theta_i\right), i = 1, \cdots, 60\right\}$. Este conjunto en unión con el polinomio formado por $\nabla^2 W$ y $\frac{\partial W}{\partial n}$ alimentan a un ajuste por mínimos cuadrados general. Este método da como resultado el conjunto de coeficientes c_i que se observan en la ecuación (2.19), véase C.VII. El resultado final se presenta en pantalla como en la figura 3.9. En esa figura se detalla el nombre de la serie de imágenes de prueba, el archivo fuente de datos, el nombre de un archivo bitácora con extensión REG. En ese reporte se muestran el valor de cada aberración y su coeficiente asociado. Existe la posibilidad de realizar diferentes ajustes e incorporar el resultado a una bitácora existente, ver figura (3.10). Para efectuar el ajuste se escoge el conjunto de datos asociados a un par de imágenes, después se escoge una bitácora nueva o se anexa el resultado a una bitácora existente. La programación de todos los módulos es el contenido

```
                    F A S E S

Diferencia de Dos Imagenes (RODDIER),
Ajuste por Minimos Cuadrados General ............    EJECUTADOS

Escritura de Zernikes en formato FAST ...........    EJECUTADA


                 Aberraciones Monocromáticas

C1  =    0.0000000000E+00     Tilt eje x
C2  =    0.0000000000E+00     Tilt eje y
C3  =    1.2988736690E+00     Desenfoque
C4  =    0.0000000000E+00     Astig 3r orden sobre el eje x
C5  =    0.0000000000E+00     Astig 3r orden a 45 del eje x
C6  =   -6.6005946358E-02     Coma 3r orden sobre el eje x
C7  =   -4.5871833359E-02     Coma 3r orden sobre el eje y
C8  =   -1.0153130159E-01     Aberración esférica de 3er orden
C9  =    0.0000000000E+00     Coma Triangular eje y
C10 =    0.0000000000E+00     Coma Triangular eje x


Tiempo transcurrido :          2.2000002861E-01   segundos
     Archivo Fuente :          H035.*DT
   Archivo Registro :          SERIE1.REG
        †† Pinche el Mouse para continuar ††
```

Figura 3.9. Coeficientes de Zernike.

del apéndice C.

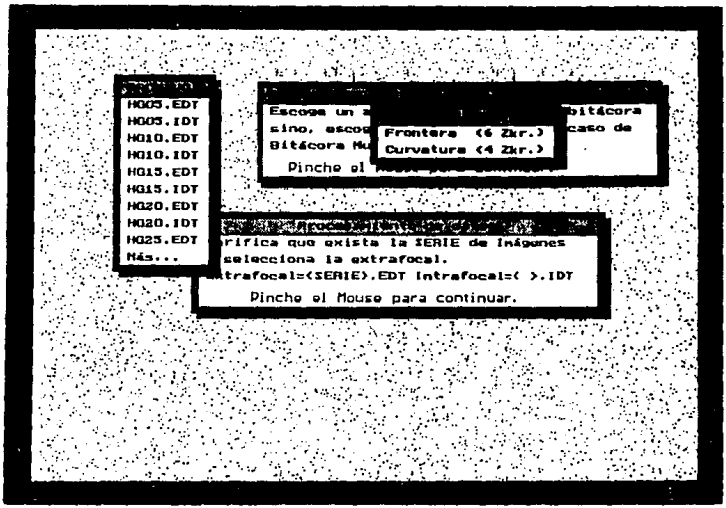

Figura 3.10. Selección de parejas de datos y bitácora para el ajuste.

CAPITULO CUATRO

Verificación Experimental

4.1 Introducción.

Para comprobar la instrumentación del método, se probó una parábola pulida en el taller de Optica del Instituto, en su centro de curvatura. La prueba se realizó utilizando un interferómetro de Twyman-Green, el cual cuenta con una cámara de televisión para digitalizar las franjas de interferencia. Las franjas digitalizadas son procesadas mediante un programa comercial de análisis que permite obtener *"mapas"* de la superficie a probar. Dicho programa efectua un ajuste por mínimos cuadrados de la superficie con un conjunto de funciones base en términos de Zernike. El software utilizado es FAST/VAI de Phase Shift Technology de Tucson, Arizona.

Posteriormente se probó, también en el centro de curvatura, la parábola utilizando el método de Roddier descrito en el capítulo anterior, como se menciona en el capitulo DOS , la salida del programa da un desarrollo de polinomios de Zernike.

Los resultados del sistema interferométrico fueron comparados con los resultados del programa.

4.2 Test Interferométrico.

El sistema de test interferométrico esta basado en un montaje de tipo de Twymann-Green[10]. En la Figura 4.1 se muestra el montaje utilizado. El haz de un laser de He-Ne de 5 mW es expandido con un objetivo de microscopio y filtrado con un pinhole de 100 micras. Un colimador forma un haz aproximadamente plano, el cual es dividido en dos partes por el divisor de haz. Una parte se refleja en un espejo plano de referencia, cuya planeidad es mejor que $\lambda/10$. La otra parte del haz se dirige hacia el espejo de prueba. El haz saliente es plano y se convierte en un haz esférico mediante la lente auxiliar, siendo ésta un objetivo fotográfico diafragmado. La finalidad es realizar el test en el centro de curvatura del espejo parábolico. El foco del objetivo debe hacerse coincidir con el centro de curvatura. Debido a que el radio de curvatura de la parábola es de 2 m y la mesa óptica de menores dimensiones, se plegó el haz por medio de un espejo auxiliar. El haz que regresa del espejo de referencia y el de la superficie de prueba se recombinan en el divisor de haz. Las franjas de interferencia producidas por ambos frentes se forman detrás del divisor. Con el fin de conseguir una buena pupila en la cámara CCD de digitalización, un objetivo de enfoque se coloca enfrente, formando una imagen del espejo en la parte sensible de la cámara. La lente auxiliar introduce sus propias aberraciones en la trayectoria del frente de onda de prueba. Por tanto es necesaria una previa calibración, se mide las aberraciones de la lente auxiliar con una superficie cóncava esférica de referencia (a $\lambda/20$) como se muestra en la figura 4.2. Las aberraciones y el frente de onda de la lente auxiliar se almacenan en

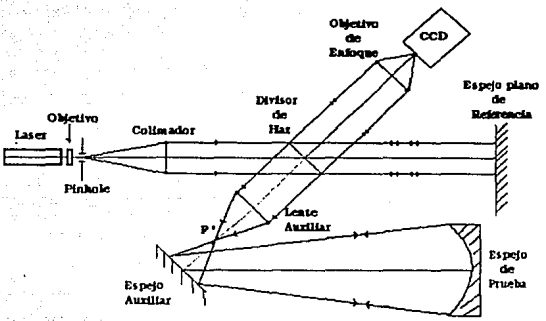

Figura 4.1. Prueba Interferométrica.

un archivo en disco y es restada su contribución del frente de onda medido con el espejo parábolico. Los términos de la prueba interferométrica se muestran en la tabla 4.1 y la reconstrucción de ese frente de onda esta contenido en la figura 4.4.

4.3 El método de Roddier.

En el capítulo DOS se explica la aplicación del método de Roddier, en él se deben de capturar imágenes intra y extrafocales simétricas al plano focal del sistema bajo prueba. Con el fin de aplicar la prueba de Roddier a la parábola del inciso precedente, se realizó el montaje óptico contenido en la figura 4.3. Una lámpara de mercurio ilumina un pinhole a través de un condensador. El haz cónico resultante se envia hacia la parábola de prueba. Un divisor de haz de tipo cubo (con defectos inferiores a $\lambda/10$), permite formar la imagen del pinhole en una dirección perpendicular al eje óptico del espejo. Sobre este plano se coloca una cámara CCD montada en un riel óptico. Esta cámara se desplaza simétricamente al plano focal, que en este caso coincide con el centro de curvatura del espejo, a fin de obtener las imágenes intrafocales y extrafocales. El parámetro f coincide con el valor del radio de curvatura mostrado en la tabla 4.3. Los valores de ℓ para cada par de imágenes también estan contenidos en la segunda columna de la misma tabla. Cada par de imágenes desenfocadas son digitalizadas y grabadas como archivos en la computadora, además de la diponibilidad de un registro de video en una videocasetera. Las aberraciones introducidas por el divisor de cubo son despreciables, ya que el número de apertura del haz es de f/12, véase el cálculo de las aberraciones del cubo en el Apéndice D. Como se explica en el capítulo TRES, cada par de imágenes son procesadas en conjunto con sus parámetros (λ, ℓ, f y D). El valor de ℓ es el desplazamiento de la cámara sobre el banco óptico con respecto al centro de curvatura. Los valores del ajuste a los términos de Zernike

se muestran en la tabla 4.2. Debido a que se consideraron cuatro pares de imágenes, cada par da un valor para cada aberración y la aberración final es el promedio de esos valores. Los valores de λ, ℓ, f, d de las imágenes involucradas están en la tabla 4.3. Con estos resultados e introduciendo los valores en el programa FAST, se hizo la reconstrucción del frente de onda, ver figura 4.5.

4.4 Resultados.

a) Con los datos experimentales hubo problemas de saturación para imágenes cercanas al foco y de sensibilidad del detector para imágenes alejadas del foco. Este hecho se refleja en una disparidad de resultados en cada par de imágenes desenfocadas.

b) Los valores obtenidos en la prueba interferométrica fueron los representados en la tabla 4.1, en ésta se excluyen los términos de tilt y desenfoque, ya que no corresponden a los medidos interferométricamente. Los valores con la prueba de Roddier, con imágenes no-saturadas, están contenidos en la tabla 4.2.

Los valores de los coeficientes de Zernike para astigmatismo son completamente diferentes en ambas pruebas, por tanto se excluyeron de la reconstrucción de los frentes de onda (interferométrico y de Roddier).

Considerando sólo tres coeficientes de aberración: dos de coma y uno de aberración esférica, se puede hacer una comparación entre una reconstrucción del frente de onda generado por la prueba de Roddier (ver figura 4.5) y el frente de onda *real* del espejo (figura 4.4), obtenido en la prueba interferométrica. El aspecto de ambos es el mismo, debido a que las aberraciones dominantes son coma y aberración esférica. Para los términos de coma c_6 y c_7 los errores relativos son de un 20% y un 27% respectivamente, mientras que en el caso de aberración esférica es del 2% entre ambas pruebas.

En el caso del astigmatismo no se obtuvieran resultados consistentes, lo cual nos hace pensar que el método sólo funciona para aberraciónes simétricas. La contribución de la diferencia de imágenes correspondiente al astigmatismo, esta dada en la orilla de las imágenes desenfocadas. En ese lugar la señal es mas intensa y satura el detector antes que el centro de la imagen. Además es difícil determinar el muestreo de la imagen en el borde, lo que puede introducir graves errores.

4.5 Conclusiones.

- El método instrumentado en esta tesis es cómodo de usar y utiliza un algoritmo interesante para reconstruir el frente de onda.
- Se cumplen entonces la especificaciones que se dieron para el método en el capítulo DOS.
- El método funciona de manera adecuada para aberraciones simétricas. Posiblemente se hace un muestreo insuficiente de los valores en la orilla de la imagen.
- Un punto fundamental en la limitación del método está en la forma de resolver la ecuación de Roddier: el Laplaciano de los Zernikes no es forzozamente ortogonal, reflejándose en una mezcla de las aberraciones. Esto fue discutido con F. Roddier personalmente.

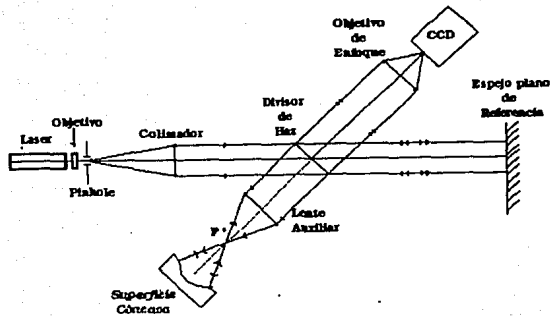

Figura 4.2. Caracterización lente auxiliar.

- Hay que hacer notar, que en la actualidad otros métodos para resolver la ecuación de Roddier son bastante largos y engorrosos, dando resultados similares a los presentados en esta tesis[11].

C_6	Coma de 3er. orden	+0.019
C_7	Coma de 3er. orden	−0.069
C_8	Aberración esférica	−0.098

Tabla 4.1. Polinomios de Zernike de la prueba interferométrica.

C_6	Coma de 3er. orden	+0.015
C_7	Coma de 3er. orden	−0.050
C_8	Aberración esférica	−0.100

Tabla 4.2. Polinomios de Zernike de la prueba de Roddier

Serie de Imágenes	ℓ	f (mm)	R	λ
135	67.5	2000	76	0.6328×10^{-3}
140	70.0	2000	76	0.6328×10^{-3}
145	72.5	2000	76	0.6328×10^{-3}
150	75.0	2000	76	0.6328×10^{-3}

Tabla 4.3. Imágenes consideradas en el ajuste con sus valores de (ℓ, f, R, λ).

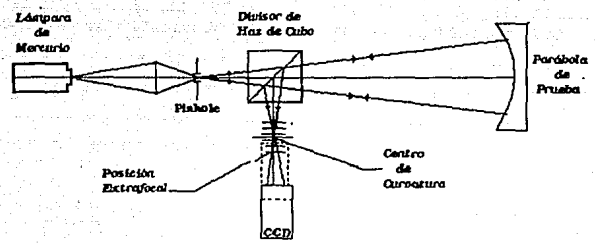

Figura 4.3. Método de Roddier.

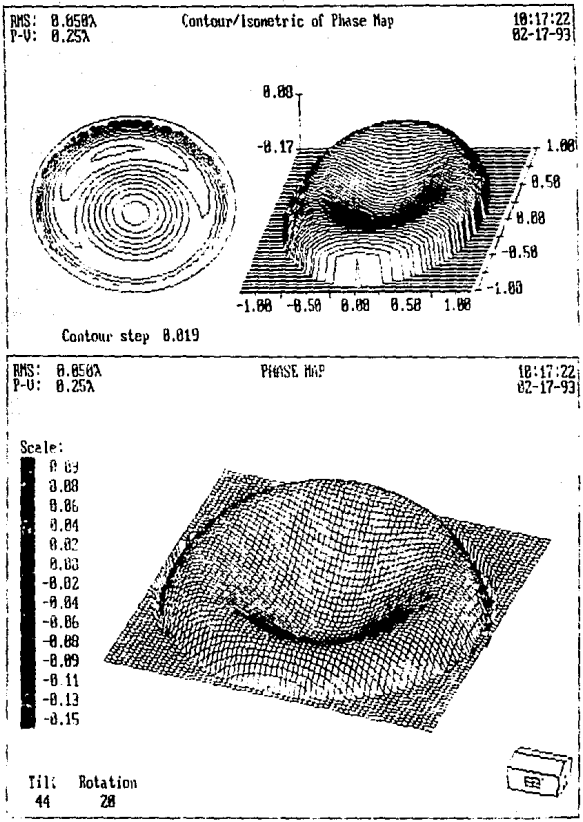

Figura 4.4. Reconstrucción de W por FAST a partir de la prueba interferométrica.

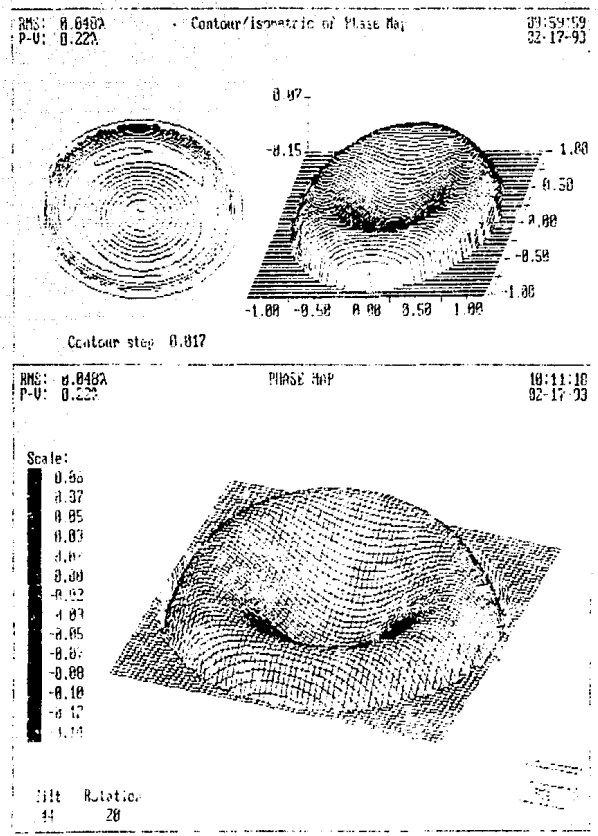

Figura 4.5. Mapa en FAST de Prueba de Roddier.

APENDICE A

Polinomios de Zernike

A.I Introducción.

El conjunta mas recomendable en la descripción de sistemas ópticos es el conjunto de los polinomios de Zernike[12]. Ejemplo del uso de estos polinomios es la descripción estadística de las aberraciones en fenómenos de turbulencia atmosférica[13]. En este apéndice solo se enumera algunas de las propiedades de los polinomios de Zernike, tales como la representación integral, fórmula generadora y la derivada de Zernike. La última sección del apéndice se dedica exclusivamente al cálculo de $\nabla^2 W$ y $\frac{\partial W}{\partial n}$ en coordenadas cilíndricas.

A.II Polinomios de Zernike.

Se define a los polinomios de Zernike como un conjunto de polinomios sobre un círculo unitario. El uso de coordenadas polares es la representación más adecuada y se refleja en el hecho de una representación en forma de producto de funciones angulares y polinomios radiales.

Los polinomios de Zernike se definen como

$$Z_j = \begin{cases} \sqrt{n+1}R_n^m(r)\sqrt{2}\cos m\theta & \text{j par, } m \neq 0 \\ \sqrt{n+1}R_n^m(r)\sqrt{2}\text{sen}m\theta & \text{j impar, } m \neq 0 \\ \sqrt{n+1}R_n^0(r) & m \neq 0 \end{cases} \tag{A.1}$$

donde

$$R_n^m(r) = \sum_{s=0}^{\frac{n-m}{2}} \frac{(-1)^s(n-s)!}{s!\,[(n+m)/(2-s)]!\,[(n-m)/(2-s)]!} r^{n-2s} \tag{A.2}$$

con n, m enteros que satisfacen $m \leq n$, $n - |m| =$ par.

De la definición A.1 se llega a una relación modal ortogonal como sigue

$$\int d^2r W(r) Z_j Z_{j'} = \delta_{jj'} \tag{A.3}$$

con la función de peso $W(r) = \begin{cases} 1/\pi & r \leq 1 \\ 0 & r > 1 \end{cases}$

Una aplicación típica de los polinomios de Zernike es la expresión de un frente de onda arbitario sobre una apertura circular de radio R. Sea $\phi(r,\theta)$ es una función arbitraria, su expansión polinomial sobre un círculo de radio R esta dada como

$$\phi(R\rho,\theta) = \sum_j a_j R_j(\rho,\theta) \tag{A.4}$$

con $\rho = \frac{r}{R}$. Los coeficientes a_j se definen como sigue

$$a_j = \frac{1}{R^2}\int d^2r W(\frac{r}{R})\phi(r,\theta)Z_j(\frac{r}{R},\theta) \qquad (A.5)$$

Los primeros polinomios desarrollados de la expresión A.5, se encuentran en la tabla A.1 junto con su aberración asociada[8].

Grado radial (m) (m)	Frecuencia Azimutal 0	1	2
1		$Z_2 = 2r\cos\theta$ tilt $Z_3 = 2r\text{sen}\theta$ (posición lat.)	
2	$Z_4 = \sqrt{3}(2r^2 - 1)$ desenfoque posición long.		$Z_5 = \sqrt{6}r^2\text{sen}2\theta$ $Z_6 = \sqrt{6}r^2\cos 2\theta$ Astigmatismo 3er. orden
3		$Z_7 = \sqrt{8}(3r^3 - 2r)\text{sen}\theta$ $Z_8 = \sqrt{8}(3r^3 - 2r)\cos\theta$ coma 3er. orden	
4	$Z_{11} = \sqrt{5}(6r^4 - 6r^2 + 1)$ aberración esférica		

Tabla A.1 Polinomios de Zernike asociados a Aberraciones monocromáticas.

A.III Derivada de Zernike.

La derivada de la parte radial de estos polinomios es posible expresarla como una función de recurrencia

$$\frac{d}{d\rho}R_n^m = n\left[R_{n-1}^{m+1} + R_{n-1}^{m-1}\right] + \frac{d}{d\rho}R_{n-2}^m \qquad (A.6)$$

La relación de recurrencia (A.6) proporciona una representación de la derivada de polinomios de Zernike como una combinación de estos mismos, es decir

$$\nabla Z_j = \sum_{j'} \delta_{jj'} Z_{j'} \qquad (A.7)$$

la matriz δ expresada en coordenadas rectangulares es de la forma

$$\delta_{jj'}^x = \int d^2\rho Z_{j'}\frac{dZ_j}{dx} \qquad (A.8)$$

y

$$\delta^y_{jj'} = \int d^2\rho Z_{j'} \frac{dZ_j}{dy} \qquad (A.9)$$

Los elementos de matriz $\delta^x_{jj'}$ y $\delta^y_{jj'}$ tienen ciertas reglas de construcción reflejadas en dos tablas contenidas en el artículo de Noll[8]. Un punto importante es la representación de la derivada de Zernike como la combinación lineal de ellos mismos, véase la ecuación (A.7) y que los primeros modos de estos polinomios representan aberraciones clásicas ópticas como en la Tabla A.1.

A.IV Cálculo de $\nabla^2\mathcal{W}$ y $\frac{\partial\mathcal{W}}{\partial n}$

La expresión del laplaciano en coordenadas cilíndricas es

$$\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial\psi}{\partial r}\right) + \frac{1}{r^2}\frac{\partial^2\psi}{\partial\theta^2} + \frac{\partial^2\psi}{\partial z^2} + \kappa^2\psi = 0 \qquad (A.10)$$

Dada la simetría existente alrededor del eje Z la expresión se reduce a

$$\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial\psi}{\partial r}\right) = \frac{1}{r}\left\{r\frac{\partial^2\psi}{\partial r^2} + \frac{\partial\psi}{\partial r}\right\} = \frac{\partial^2\psi}{\partial r^2} + \frac{1}{r}\frac{\partial\psi}{\partial r} \qquad (A.11)$$

Los polinomios de Zernike tienen la forma $Z_n(r,\theta) = r^n\cos n\theta$ y los términos considerados son

$Z_1 = 2r\cos\theta c_1$	tilt (posición lateral).
$Z_2 = 2r\mathrm{sen}\theta c_2$	tilt (posición lateral).
$Z_3 = \sqrt{3}(2r^2-1)c_3$	desenfoque (posición longitudinal).
$Z_4 = \sqrt{6}r^2\mathrm{sen}2\theta c_4$	Astigmatismo 3er. orden.
$Z_5 = \sqrt{6}r^2\cos 2\theta c_5$	Astigmatismo 3er. orden.
$Z_6 = \sqrt{8}(3r^3-2r)\mathrm{sen}\theta c_6$	Coma 3er. orden.
$Z_7 = \sqrt{8}(3r^3-2r)\cos\theta c_7$	Coma 3er. orden.
$Z_8 = \sqrt{5}(6r^4-6r^2+1)c_8$	Aberración esférica.

La parte de curvatura $\nabla^2\mathcal{W}$ da el resultado de

$$\begin{aligned}
\nabla^2 Z_1 &= 0\\
\nabla^2 Z_2 &= 0\\
\nabla^2 Z_3 &= 8\sqrt{3}c_3\\
\nabla^2 Z_4 &= 0\\
\nabla^2 Z_5 &= 0\\
\nabla^2 Z_6 &= 24\sqrt{8}c_6 R\cos\theta\\
\nabla^2 Z_7 &= 24\sqrt{8}c_7 R\mathrm{sen}\theta\\
\nabla^2 Z_8 &= 24\sqrt{5}c_8\left\{4R^2-1\right\}
\end{aligned}$$

Los términos de orilla se calculan por medio de la siguiente expresión.

$$\left.\frac{\partial \mathcal{W}(r,\theta)}{\partial n}\right|_{R=1} = \left.\frac{\partial Z_1}{\partial r}\right|_{R=1} \qquad j = 1,\ldots 8 \qquad (A.12)$$

y dan el siguiente resultado

$$\left.\frac{\partial Z_1}{\partial r}\right|_{R=1} = 2c_1 \cos\theta$$

$$\left.\frac{\partial Z_2}{\partial r}\right|_{R=1} = 2c_2 \mathrm{sen}\theta$$

$$\left.\frac{\partial Z_3}{\partial r}\right|_{R=1} = 4\sqrt{3}c_3$$

$$\left.\frac{\partial Z_4}{\partial r}\right|_{R=1} = 2\sqrt{6}c_4 \cos^2\theta$$

$$\left.\frac{\partial Z_5}{\partial r}\right|_{R=1} = 2\sqrt{6}c_5 \mathrm{sen}^2\theta$$

$$\left.\frac{\partial Z_6}{\partial r}\right|_{R=1} = 7\sqrt{8}c_6 \cos\theta$$

$$\left.\frac{\partial Z_7}{\partial r}\right|_{R=1} = 7\sqrt{8}c_7 \mathrm{sen}\theta$$

$$\left.\frac{\partial Z_8}{\partial r}\right|_{R=1} = 12\sqrt{5}c_8$$

APENDICE B

Métodos Numéricos.

B.I Método de mínimos cuadrados general.

Es una práctica necesaria el condensar y procesar datos a partir de un conjunto de observaciones, por medio de un ajuste a un modelo dependiente de parámetros variables. El modelo puede estar compuesto de funciones tales como conjuntos de polinomios o funciones gaussianas proporcionando el ajuste una serie de coeficientes apropiados. La forma de hacerlo en todos los casos es la misma: se escoge o se diseña una función determinada que mida la relación entre los datos y el modelo con una particular elección de parámetros. Los parámetros del modelo se ajustan para alcanzar un mínimo de esa función estableciendo *parámetros de ajuste*. El proceso de ajuste se reduce a un problema de minimización en varias dimensiones.

Existen aspectos inherentes que van más alla de solo encontrar los parámetros del mejor ajuste. Los datos en general no son exactos, estando sujetos a errores de medición (ruido en el contexto de procesamiento de señales), así datos típicos nuncan se ajustarían al modelo escogido, aún cuando éste sea adecuado. Surge entonces la necesidad de una prueba de *bondad de ajuste* en las técnicas estadísticas.

Se ajustan N puntos (x_i, y_i), $i = 1, \ldots, N$ a un cierto modelo con M parámetros ajustables $y = 1, \ldots, M$. El modelo predice una relación funcional entre la variable independiente (medición experimental) y las variables dependientes.

$$y(x) = y(x; a_1 \ldots a_m) \qquad (B.1)$$

donde la dependencia de los parámetros es indicada explícitamente en el miembro derecho.

El método más usual para obtener los valores ajustados del conjunto a_j's es el de mínimos cuadrados, el cual se basa en *estimadores de máxima similitud*. Para ello debe buscarse los valores que maximizen dicha similitud.

Considerando las ideas anteriores expresadas en una expresión de mínimos cuadrados, sería de la forma

$$\min\{a_i\}_1^M: \qquad \sum_{i=1}^{N}[y_i - y(x_i; a_1 \ldots a_M)]^2 \qquad (B.2)$$

La expresión (B.2) es una estimación de máxima similitud de los parámetros ajustados siempre y cuando los errores en la medición sean independientes y distribuidos normalmente con una desviación constante. El modelo no contempla ninguna suposición sobre la linealidad de $y(x; a_1, \ldots a_M)$.

Si cada punto tiene una desviación propia, la ecuación (B.2) se transforma en

$$\chi^2 = \sum_{i=1}^{N} \left(\frac{y_i - y(x_i; a_1 \ldots a_M)}{\sigma_i} \right)^2 \qquad (B.3)$$

denominada *distribución de χ-cuadrada* o *ajuste por mínimos cuadrados pesados*. En el caso de ajustar por medio de un polinomio compuesto de m funciones a un conjunto de datos (x_i, y_i), (B.3) se transforma en

$$y(x) = \sum_{k=1}^{M} a_k X_k(x) \qquad (B.4)$$

con $X_i(x), \ldots, X_M(x)$ funciones fijas, llamado también *conjunto base* de x. El conjunto básico puede ser constituido por funciones lineales. El término lineal reside en la dependencia del modelo con los parámetros a_k.

El modelo a ser resuelto tiene la forma

$$\chi^2 = \sum_{i=1}^{N} \left(\frac{y_i - \sum_{k=1}^{M} a_k X_k(x_i)}{\sigma_i} \right)^2 \qquad (B.5)$$

siendo σ_i la incertidumbre (desviación estandar) del i-ésimo dato, si se da el caso de desconocerla se debe fijar $\sigma = 1$.

La técnica usada para minimizar χ^2 se puede ver en capítulo 14 del Press[14]. Sea A una matriz de $N \times N$ construida a partir de M funciones base evaluadas en N abcisas x_i con N desviaciones σ_i de la forma

$$A_{ij} = \frac{X_i(x_i)}{\sigma_i} \qquad (B.6)$$

La matriz A es la matriz del problema de ajuste y por lo general tiene más renglones que columnas $N \geq M$, resultado de la existencia de un mayor número de datos que de parámetros, véase la figura B.1.

Es necesario también definir al vector b de longitud N como

$$b_i = \frac{y_i}{\sigma_i} \qquad (B.7)$$

y al vector a cuyas componentes $a_1, \ldots a_M$ son los parámetros del ajuste.

La solución se basa en la minimización de χ^2, para ello debe establecerse m ecuaciones que cumplan la siguiente condición

$$0 = \sum_{i=1}^{N} \frac{1}{\sigma_i^2} \left[y_i - \sum_{j=1}^{M} a_j X_j(x_i) \right] X_k(x_i) \quad k = 1, \ldots, M \qquad (B.8)$$

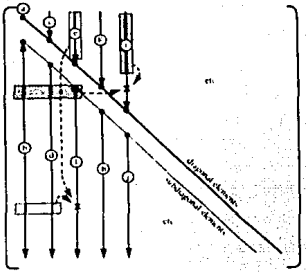

Figura B.1. Algoritmo de Crout para la descomposición LU de una matriz. Los elementos de la matriz original son modificados en el orden indicado por las letras minúsculas: a,b, c, etc.

Cambiando el orden de las sumas, la relación anterior puede expresarse en una notación matricial

$$\sum_{j=1}^{M} \alpha_{kj} a_j = \beta_k \tag{B.9}$$

$$\alpha_{kj} = \sum_{i=1}^{N} \frac{X_j(x_i) X_k(x_i)}{\sigma_i^2}$$

que equivale a $[\alpha] = \mathbf{A}^T \cdot \mathbf{A}$ matriz de $N \times M$, y

$$\beta_k = \sum_{i=1}^{M} \frac{y_i X_k(x_i)}{\sigma_i^2} \tag{B.10}$$

o $[\beta] = \mathbf{A}^t \cdot \mathbf{b}$ vector de longitud M.

Las ecuaciones (B.7) y (B.8) constituyen las ecuaciones normales de mínimos cuadrados. La resolución de estas ecuaciones da el valor del vector **a** mediante la aplicación del método LU, descrito en la siguiente sección. El método de descomposición LU sobrepasa en un factor de tres al método de eliminación gaussiana en su ejecución.

Reescribiendo en notación matricial las ecuaciones normales

$$[\alpha] \cdot \mathbf{a} = [\beta] \qquad \text{ó} \qquad (\mathbf{A}^T \cdot \mathbf{A}) \cdot \mathbf{a} = \mathbf{A}^T \cdot \mathbf{b} \tag{B.11}$$

con una variancia asociada σ_j estimada cuya expresión es

$$\sigma^2(a_j) = \sum_{i=1}^{N} \sigma_i^2 \left(\frac{\partial a_j}{\partial y_i}\right)^2 \qquad (B.12)$$

Dado que a_{ik} es independiente de y_i entonces

$$\frac{\partial a_j}{\partial y_i} = \sum_{k=1}^{M} \frac{C_{jk} X_k(x_i)}{\sigma_i^2} \qquad (B.13)$$

se puede tener la siguiente expresión

$$\sigma^2(a_j) = \sum_{k=1}^{M} \sum_{l=1}^{M} C_{jk} C_{jl} \left[\sum_{i=1}^{N} \frac{X_k(x_i) X_l(x_i)}{\sigma_i^2}\right] \qquad (B.14)$$

Dado que $[\alpha]$ es la matriz inversa de $[C]$, la ecuación (B.14) se reduce a

$$\sigma^2(a_j) = C_{jj} \qquad (B.15)$$

Esto indica que los elementos de la diagonal son las variancias (cuadrado de la incertidumbre) de los parámetros de ajuste a. Es importante mencionar que la solución de un problema de mínimos cuadrados por medio de ecuaciones normales es susceptible a errores de redondeo.

B.II Descomposición LU.

Un conjunto de ecuaciones lineales algebraicas es de la forma

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n &= b_1 \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n &= b_2 \\ &\vdots \\ a_{n1}x_1 + a_{n2}x_2 + \ldots + a_{nn}x_n &= b_n \end{aligned} \qquad (B.16)$$

con N incógnitas x_i relacionadas con M ecuaciones. Si $N = M$ existe un número similar de ecuaciones e incógnitas y una buena oportunidad de obtener una solución única para el conjunto x_j. Analíticamente no existe una solución única, si una o más ecuaciones de las M existentes es una combinación lineal, a esta condición se le llama *degeneración* de renglón o columna. A un conjunto de ecuaciones degenerado se denomina *singular*. Es de nuestro interés las matrices no-singulares.

Numéricamente existen dos causas que provocan anomalías en la resolución del problema.

- La existencia de errores de redondeo que se reflejan en ecuaciones linealmente dependientes en alguna etapa del proceso. El proceso debe notificarlo al usuario.

- Errores de redondeo acumulados durante el proceso escondiendo la solución real. Esta situación sucede con N demasiado grande , no siendo imputable al algoritmo. Se puede detectar mediante una sustitución hacia atrás en el conjunto de ecuaciones.

Un conjunto de ecuaciones lineales con N entre 20 y 50 puede ser resuelto con una precisión sencilla (32 bit en representación flotante) sin recurrir a métodos sofisticados, siempre y cuando el sistema no se aproxime demasiado a ser singular. Con una precisión doble N puede ser cercana a 100, donde la limitante es el tiempo no la precisión. Cuando se tiene un sistema singular con N pequeñas ($N = 10$) conviene utilizar el método de descomposición (SVD) *

En la sección anterior se planteó un sistema como el de (B.11)

$$\mathbf{A} \cdot \mathbf{x} = \mathbf{b} \tag{B.17}$$

donde $\mathbf{A}$ es la matriz de coeficientes y $\mathbf{b}$ es el vector columna cuya representación es

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1N} \\ a_{21} & a_{22} & \dots & a_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ a_{M1} & a_{M2} & \dots & a_{MN} \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_M \end{pmatrix} \tag{B.18}$$

En el caso de $M < N$ o $M = N$ y además hay degeneración, no existe una solución única o una infinidad de soluciones para el vector $\mathbf{x}$. Cuando $M > N$ no existe el vector solución $\mathbf{x}$, entonces se dice que el sistema esta sobredeterminado. Sin embargo se puede buscar una solución *cercana* que satisfaga a todas las ecuaciones. La cercanía es en el sentido de mínimos cuadrados *i.e.* que la suma del cuadrado de las diferencias entre el lado derecho y el izquierdo de la ecuación (B.17) sea un mínimo, de esta manera el sistema sobredeterminado se reduce a un problema con solución vía mínimos cuadrados.

El conjunto reducido de ecuaciones de $N \times N$ se escribe como

$$(\mathbf{A}^T \cdot \mathbf{A}) \cdot \mathbf{x} = (\mathbf{A}^T \cdot \mathbf{b}) \tag{B.19}$$

donde $\mathbf{A}^T$ es la transpuesta de $\mathbf{A}$. Estas ecuaciones se les conoce como las **ecuaciones normales** del problema de mínimos cuadrados.

Se puede escribir a la matriz $\mathbf{A}$ como el producto de dos matrices

$$\mathbf{L} \cdot \mathbf{U} = \mathbf{A} \tag{B.20}$$

donde la matriz $\mathbf{L}$ es triángular inferior (solo hay elementos en y bajo de la diagonal) y $\mathbf{U}$ es una triángular superior. En el caso de una matriz $\mathbf{A}$ de 4×4, sería de la forma

* Véase la sección 2.9 del Press[14].

$$\begin{pmatrix} \alpha_{11} & 0 & 0 & 0 \\ \alpha_{21} & \alpha_{22} & 0 & 0 \\ \alpha_{31} & \alpha_{32} & \alpha_{33} & 0 \\ \alpha_{41} & \alpha_{42} & \alpha_{43} & \alpha_{44} \end{pmatrix} \begin{pmatrix} \beta_{11} & \beta_{12} & \beta_{13} & \beta_{14} \\ 0 & \beta_{22} & \beta_{23} & \beta_{24} \\ 0 & 0 & \beta_{33} & \beta_{34} \\ 0 & 0 & 0 & \beta_{44} \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{pmatrix} \qquad (B.21)$$

La descomposición anterior sirve para resolver un sistema como el de (B.17), quedando como

$$\mathbf{A} \cdot \mathbf{x} = (\mathbf{L} \cdot \mathbf{U}) \cdot \mathbf{x} = \mathbf{L} \cdot (\mathbf{U} \cdot \mathbf{x}) = \mathbf{b} \qquad (B.22)$$

donde primero se resuelve $\mathbf{L} \cdot \mathbf{y} = \mathbf{b}$ y después $\mathbf{U} \cdot \mathbf{x} = \mathbf{y}$

La ventaja de la descomposición del sistema en dos ecuaciones es el uso de matrices triangulares, cuya resolución es trivial mediante el método de sustitución hacia atrás como sigue

$$y_1 = \frac{b_1}{\alpha_{11}} \qquad y_i = \frac{1}{\alpha_{ii}} \left[b_i - \sum_{j=1}^{i-1} \alpha_{ij} y_j \right] \qquad i = 2, 3, \ldots, N \qquad (B.23)$$

mientras que la segunda matriz

$$x_N = \frac{y_N}{\beta_{NN}} \qquad x_i = \frac{1}{\beta_{ii}} \left[y_i - \sum_{j=i+1}^{N} \beta_{ij} x_j \right] \qquad i = N-1, N-2, \ldots, 1 \qquad (B.24)$$

Una vez descompuesta A en un sistema LU, es posible resolver el problema con diferentes b's en una sola ocasión a diferencia del método de Gauss-Jordan.

Para resolver L y U dada una matriz A se utiliza el *algoritmo de Crout* como sigue

- Se fija $\alpha_{ii} = 1 \quad i = 1, \ldots, N$.
- Para cada $j = 1, 2, 3, \ldots, N$ se realizan dos pasos

a) $\forall\, i = 1, 2, \ldots, j$ se resuelve β_{ij}

$$\beta_{ij} = a_{ij} - \sum_{k=1}^{i-1} \alpha_{ik} \beta_{kj} \qquad (B.25)$$

b) $\forall\, i = j+1, j+2, \ldots, N$ se resuelve α_{ij}

$$\alpha_{ij} = \frac{1}{\beta_{ij}} (a_{ij} - \sum_{k=1}^{j-1} \alpha_{ik} \beta_{kj}) \qquad (B.26)$$

las ecuaciones (B.25) y (B.26) deben aplicarse antes de la siguiente j.

El método de Crout lleva una matriz combinada de α's y β's

$$\begin{pmatrix} \beta_{11} & \beta_{12} & \beta_{13} & \beta_{14} \\ \alpha_{21} & \alpha_{22} & \beta_{23} & \beta_{24} \\ \alpha_{31} & \alpha_{32} & \alpha_{33} & \beta_{34} \\ \alpha_{41} & \alpha_{42} & \alpha_{43} & \beta_{44} \end{pmatrix} \tag{B.27}$$

Un punto importante a considerar es el pivoteo, *i.e.* la selección adecuada para el elemento pivote de la división en la ecuación (B.26), que es un punto crucial en el algoritmo de Crout. Un pivoteo parcial (intercambio de renglones) puede ser suficiente con el fin de tener un método más estable.

La inclusión del pivoteo no es obvia en el algoritmo de Crout. Su consideración se nota en la ecuación (B.25) en el caso de $i = j$ y también se observa en (B.26) excepto por la división; en ambos casos el límite superior de la suma es $k = j-1(= i-1)$. Para mejorar el pivoteo se escoge el elemento más grande de la diagonal β o elemento pivote calculándose todas las divisiones en masa. Esta mejora se denomina *algoritmo de Crout con pivoteo parcial.* Un avance adicional es posible si en un inicio se busca el elemento más grande en cada renglón y conforme se avanza en cada etapa, se *escalan* todas las ecuaciones para hacer que su máximo coeficiente sea la unidad, convirtiéndose esta operación en un pivoteo implícito.

APENDICE C

Código Fuente de Programación

C. I Programa Principal (RODDIER.PAS).

```
{$I HEAD}

Program Roddier(input,output);
(* Mayo 8 '90
(* Este programa hace el ensamblaje de todos los modulos necesarios para realizar el metodo de Roddier
    Direccion de:          Salvador Cuevas C.
    Asesoria de :          Fernando Angeles

    Programción : Fierro Pacheco Salvador
                                                                                Instituto de Astronomia UNAM
    Utiliza tarjeta Genoa,una 286,tajeta digitalizadora Dt2858, camara de video monitor de Color con tarjeta Genoa en algunos de las
    unidades    ******)

    Uses            Crt,
                    Graph,
                    Mouse,
                    Menu,
                    FoolGraf,BglDriv,BglFont,Tipo,
                    Bk,         {Pone el fondo}
                    Ayuda,      {contiene menus de ayuda  e instrucciones          }
                    Menudo,     {Contiene algunas utilerias                  }
                    Iris,       {Esta unidad captura la imagen procedente de VCR o video}
                    Scan2,      {escoge los puntos en la imagen a utilizar en el ajuste }
                    mrdd3,      {hace I_1-I_2/I_1+I_2                                   }
                    Disp,       {despliega el contenido de una imagen          }
                    Zeit;       {calcula el tiempo transcurrido                }
    Const
        Off = False;    On = True;
    Type
        variable=String[12];

    Var             serie1,ss1,ss2,dat,
                    intra,extra,Bita,Fuente                         :variable;
                    respuesta,Car                                   :Char;
                    Op,Opc,miraX,miraY,I,FullFrt,
                    numvar,ErrCode,c,Gd,Gm                          :Integer;
                    registro                                        :Text;
                    Existencia,Clase,Cursor                         :Boolean;
                    Polinomio,SigPol    :RealArrayMA;
                    Elapsed                                         :Real;
    Procedure Soni;
    Begin
        Sound(400);
        Delay(100);
        NoSound;
    End;

    Function TIPOFILE(Wort:ss):Boolean.
    Var
      st1 : String[12];
      P   : Integer;

    Begin
```

```
        P:=Pos('.',Wort);
        St1:=Copy(Wort,P+1,3);
        If St1='IMI' then TipoFile:=True
        Else TipoFile:=False;
    End;

    Function Recorta(ss:String) : String;
    Var P: Integer;
    Begin
        P:=Pos('.',ss);
        Recorta:=Copy(ss,1,P-1);
    End;

Begin                                   {Programa principal}
        ErrCode:=RegisterBGIdriver(@EGAVGADriverProc); ErrCode:=RegisterBGIfont(@GothicFontProc  );
        ErrCode:=RegisterBGIfont(@SansSerifFontProc); ErrCode:=RegisterBGIfont(@SmallFontProc   );
        ErrCode:=RegisterBGIfont(@TriplexFontProc  );
        Repeat
                SetGraph;
                Fondo;
                SetTextStyle(DefaultFont,0,1);
                SetMenu('CAPTURA');
                SetMenu('DESPLIEGA');
                SetMenu('MUESTREA');
                SetMenu('CONSTANTES');
(*                SetMenu('RODDIER, FITTING & FAST');  *)
                SetMenu('PROCESAMIENTO DE DATOS');

                SetMenu('SALIDA');
                Op:=MenuTo(200,150,'MENU PRINCIPAL');
                CloseBoard;
                HidemouseCursor;
                Case Op Of
                1: Begin                                {+++++ Menú de CAPTURA  ++++++++++}
                        Ayuda11;
                        Repeat until AnyMouseKeyPressed; Ss1:=PickFile(100,100,'*.IM7');
                      CloseBoard;
                      If Ss1=' then
                            Begin
                                    HideMouseCursor; DrawBoard(170,220,270,236,'CAPTURA:');
                                      MoveTo(8,4);
                                            Serie1:=AskAlfa(8,'PRUF');
                                    CloseBoard;
                                    SetMenu('INTRAFOCAL');
                                    SetMenu('EXTRAFOCAL');
                                      Op:=MenuTo(270,240,'TIPO DE IMAGEN:');
                                    CloseBoard;
                                    HideMouseCursor;
                                    If Op=1 then
                                            serie1:= serie1+'.IMI'
                                    Else
                                            serie1:= Concat(serie1,'.IME');
                            End
                      Else Serie1:=Ss1;
                      SetMenu('On');
                      SetMenu('Off');
                        OpC:=MenuTo(270,240,'CURSOR:');
                      CloseBoard;
                      HideMouseCursor;
                      If OpC=1 then Cursor:=Off
                                    Else Cursor:=On;          {instrucciones}
                      HideMouseCursor;
                      Ayuda12;
                      Repeat until AnyMouseKeyPressed;
                      CloseBoard;
                      HideMouseCursor;
                      MiraX:=0;  MiraY:=0;
                      Captura(serie1,Cursor,miraX,miraY);
```

```
            End;
                    (****** En esta etapa DESPLIEGA que contiene la imagen                    ****)
        2: Begin
                    serie1:=PickFile(100,100,'*.IM');
                    HideMouseCursor;
                    if serie1<>'' then
                    Begin
                        Ayuda21;
                        Repeat until AnyMouseKeyPressed;
                        CloseBoard;
                        HideMouseCursor;
                        Show(serie1);
                    End;
            End;

(******* En esta etapa se MUESTREA a la imagen con el fin de obtener un ****) (******* número fijo de puntos.
****)
        3: Begin
                    c:=0;
                    Repeat
                        serie1:=PickFile(100,100,'*.IM7');
                        HideMouseCursor;
                        If serie1<>'' then
                        Begin
                                Clase:=TipoFile(serie1);  (Var con PickFile) Scan(serie1,Clase);    {Se muestrea la
                                imagen rotada}
                        End;
                        Inc(c);
                    until c=1;
            End;
(*****       Aqui se ingresan las CONSTANTES de la formula de Roddier  ***)
        4. Begin
                        DrawBoard(170,220,270,236,'CONSTANTE ');
                          MoveTo(8,4);
                          Serie1:=AskAlfa(8,'RD');
                        CloseBoard;
                        HideMouseCursor;
                        Cte(Serie1);   { sin extension MRDD3  }
            End;
(******      Aqui se hace el FITTING de la Imagen                    ************)
        5: Begin
                      Ayuda51;
                      Repeat until AnyMouseKeyPressed; Ss1:=PickFile(240,240,'*.?DT');
                    Dat:=Recorta(Ss1);
                    CloseBoard;
                    HideMouseCursor;
                    If Ss1<>'' then
                    Begin
                        Ayuda52;
                          Repeat until AnyMouseKeyPressed, Bita:=PickFile(100,100,'*.REG');
                          SetMenu('Frontera  (6 Zkr.)');
                          SetMenu('Curvatura (4 Zkr.)'); FullFit:=MenuTo(270,240,'AJUSTE');
                          CloseBoard;
                        CloseBoard;      {Ayuda52}
                        HideMouseCursor;
                        If Bita<>'' then
                                Begin
                                                                                Existencia:=Off;
                                                                                Bita:=dat+'.REG';
                                End
                        Else Existencia:=On;
                        ClockOn;

                        METODO(dat,Polinomio,SigPol,FullFit),
                                                                                {On 10 var, Off 4 var
                        MRDD3} ClockOff(Elapsed);
                        Son;
                        Assign(registro,Bita);
```

```
                    If Existencia then Append(registro)
                    Else Rewrite(registro);
                    Writeln(registro); Writeln(registro,':5;'Coeficientes de Zernike'); Writeln(registro);
                    Writeln(registro,'Imágenes fuente ',dat+'.IDT y .EDT'); Writeln(registro);
                    For i:=1 to 10 Do
                            Writeln(registro,'C_',i,' = ',Polinomio[i]:7:5,
                                    ' ',TipoAbe(i));
                    Writeln(registro);
                    Writeln(registro,'Tiempo transcurrido: ',Elapsed:7:5,' segundos'); Close(registro);
                    Fuente:=dat+'.DT';
                    Mascarilla(Polinomio,Fuente,Bita,Elapsed);
                    Aviso;              {escribe la sol. onScreen}
                    Repeat until AnyMouseKeyPressed;
                  End;  {Se1}
                  HideMouseCursor;
            End;  {Op 5}
                    {Case Op}
        End;
    HideMouseCursor;
    Until Op=6;
    HideMouseCursor;
    SetGraphMode(VGAHi);
    TextMode(CO80);
End.  {Programa principal}
```

C.II Fondo del módulo principal (Unidad Bk).

```
{$I HEAD}

Unit Bk;
INTERFACE
Uses Tipo,Menudo,Menu;
Type
  Cadena  = String[12];
Var
  Solucion : RealArrayMA;
  I        : Integer;

Procedure FONDO;
Procedure Mascarilla(Solucion:RealArrayMA;
            Fuente,Bitacora:Cadena;
            Duracion:Real);
Function TipoAbe(Indice:Integer):String;

IMPLEMENTATION
Uses Crt,Graph;
Procedure Fondo;
Var
  i:Integer;
  Ancho,Alto,Leyenda:Word;
Begin
   SetFillStyle(InterLeaveFill,3);      {modo 640x480}
   Bar(0,0,GetMaxX,GetMaxY);
   Rectangle(0,0,GetMaxX,GetMaxY);
   Rectangle(25,25,GetMaxX-25,GetMaxY-25);
   SetViewPort(25+1,25+1,GetMaxX-26,GetMaxY-26,True);
   ClearViewPort;
   For i:=0 to 10000 Do
      PutPixel(5+Random(640-5),5+Random(480-5),Random(15)+1);
   Ancho:=TextWidth('r');
   Alto:=TextHeight('r');
   SetTextStyle(TriplexFont,0,5);
   leyenda:=GetMaxX-TextWidth('EL METODO DE RODDIER');
   OutTextXY((leyenda div 2)-10,20,
            'EL METODO DE RODDIER');
   SetTextStyle(SmallFont,0,6);
   Alto:=TextHeight('r');
   MoveTo(26,375);
   OutText('por: Salvador Fierro Pacheco');
```

```
    MoveTo(26,375+alto);
    OutText('dirigido por: Salvador Cuevas C.');
    MoveTo(26,375+2*Alto);
    OutText('bajo la asesoria de: Fernando Angeles U.');
    SetTextStyle(TriplexFont,1,2);
    MoveTo(550,130);
    OutText('Inst. de Astronomia');
End;

Procedure Mascarilla;
Var
  Ancho,Leyenda  : Integer;

Procedure Sonl;
Begin
    Sound(400);
    Delay(100);
    NoSound;
End;

Begin
{   SetFillStyle(InterLeaveFill,3) }
    SetFillStyle(EmptyFill,3);      {modo 640x480}
    Bar(0,0,GetMaxX,GetMaxY);
    Rectangle(0,0,GetMaxX,GetMaxY);
    Rectangle(20,20,GetMaxX-20,GetMaxY-20);

    SetBkColor(LightGray);
{    SetBkColor(White);}
    SetColor(Blue);
    SetTextStyle(TriplexFont,0,4);
    leyenda:=GetMaxX-TextWidth('F A S E S');
    OutTextXY((leyenda div 2)-10,40,'F A S E S');
    SetTextStyle(DefaultFont,HorizDir,1);
    OutTextXY(80,100,'Diferencia de Dos Imagenes (RODDIER),'),
    OutTextXY(80,120,'Ajuste por Minimos Cuadrados General .............');
    OutTextXY(500,120,'EJECUTADOS');
    Sonl;
    OutTextXY(80,150,'Escritura de Zernikes en formato FAST ..........');
    OutTextXY(500,150,'EJECUTADA');
    Sonl;
    OutTextXY(100,400-15,'Tiempo transcurrido :');
    OutTextXY(350,400-15,Trans(Duracion_Real));
    OutTextXY(500,400-15,'segundos');
    OutTextXY(140,400,'Archivo Fuente :');
    OutTextXY(350,400,Fuente);
    OutTextXY(125,415,'Archivo Registro :'),
    OutTextXY(350,415,Bitacora);
    Sonl;
    OutTextXY(200,195,'Aberraciones Monocrom ticas');
    Ancho:= TextWidth('C');
    For i:=1 to 10 Do
      Begin
        OutTextXY(110,220+12*i,'C');
        OutTextXY(110+Ancho,224+12*i,Trans(i_Integer));
        OutTextXY(140,220+12*i,' =');
        OutTextXY(190,220+12*i,Trans(Solucion[i]_Real));
        OutTextXY(350,220+12*i,TipoAbe(i));
      End;
    OutTextXY(175,430,'!! Pinche el Mouse para continuar !!');
End;

Function TipoAbe;
{Escribe que tipo de aberracion le corresponde a es termino}
Var
  msg : String;

Begin
    Case indice of
```

```
      1: msg:=' Tilt eje x';
      2: msg:=' Tilt eje y';
      3: msg:=' Desenfoque';
      4: msg:=' Astig 3r orden sobre el eje x';
      5: msg:=' Astig 3r orden a 45 del eje x';
      6: msg:=' Coma 3r orden sobre el eje x';
      7: msg:=' Coma 3r orden sobre el eje y';
      8: msg:=' Aberraci¢n esf¡rica de 3er orden';
      9: msg:=' Coma Triangular eje y';
     10: msg:=' Coma Triangular eje x';
    End;
    TipoAbe:= msg;
End;

End.
```

C.III Ventanas de Ayuda e Instrucciones (Unidad Ayuda).

```
Unit Ayuda;
(**** Esta unidad contiene todos los menus de ayuda e instrucciones del
    principal RODDIER2                                          ****)
INTERFACE
  Procedure Ayuda11;
  Procedure Ayuda12;

  Procedure Ayuda21;

  Procedure Ayuda51;
  Procedure Ayuda52;

IMPLEMENTATION
USES  Graph,
    FoolGraf,BgiDriv,BgiFont,Menu;
Procedure Ayuda11;
Begin
    DrawBoard(170,220,500,290,'OPCIONES:');
    MoveTo(8,4);
    OutText(' Puedes escoger un archivo existente');
    MoveTo(8,20);
    OutText('para reescribirlo o escoge un rengl¢n');
    MoveTo(8,36);
    OutText('vacio como Archivo Nuevo (sin extension).');
    MoveTo(8,52);
    SetTextStyle(SmallFont,0,5);
    OutText(' !! Pinche el Mouse para continuar !!');
End;
Procedure Ayuda12;
Begin
    DrawBoard(170+10,220+15,500+10,270+15,'INSTRUCCIONES:');
    MoveTo(8,4);
    OutTeed('1) <Space> Entras a modo Cursor (On/Off).');
    MoveTo(8,20);
    OutTeed('2) <RightKeyMouse> Adquisici¢n de la imagen.');
    MoveTo(8,36);
    SetTextStyle(SmallFont,0,5);
    OutText(' Pinche el Mouse para continuar. ');
End;
Procedure Ayuda21;
Begin
    DrawBoard(170,220,510,330,'DESCRIPCION:');
    MoveTo(8,4);
    OutText('Despliega la imagen en 5 modos:');
    MoveTo(9,20);
    OutText(' 1) Falso Color (Pausa).');
    MoveTo(9,36);
    OutText(' 2) Rojo 3) Verde 4) Azul.');
    MoveTo(9,52);
    OutText(' 5) 16 Tonos de Gris.');
```

```
    MoveTo(9,68);
    OutText(' <AnyKeyMouse> Termina el Despliegue.');
    MoveTo(8,91);
    SetTextStyle(SmallFont,0,5);
    OutText('   Pinche el Mouse para continuar.');
 End;
 Procedure Ayuda51;
 Begin
    DrawBoard(170,220,525,290,'Procesamiento de datos');
    MoveTo(8,4);
    OutText('Verifica que exista la SERIE de lm genes');
    MoveTo(8,20);
    OutText('y selecciona la extrafocal.');
    MoveTo(8,36);
    OutText('Extrafocal=<SERIE>.EDT Intrafocal=< >.IDT ');
    MoveTo(8,52);
    SetTextStyle(SmallFont,0,5);
    OutText('   Pinche el Mouse para continuar.');
 End;
 Procedure Ayuda52;
 Begin
    DrawBoard(170,220,520,290,'Bit cora (?.REG)');
    MoveTo(8,4);
    OutText('Escoge un archivo existente como bit cora ');
    MoveTo(8,20);
    OutText('sino, escoge un renglón vacío en caso de');
    MoveTo(8,36);
    OutText('Bit cora Nueva.');
    MoveTo(8,52);
    SetTextStyle(SmallFont,0,5);
    OutText(' Pinche el Mouse para continuar. ');
 End;

 END.
```

C.IV Utilerías varias (Unidad Memudo).

```
{$I HEAD}

UNIT Menudo;

(*Este unidad despliega una ventana de ayuda para el usuario con tres
  opciones.)
  Se necesitan estas instrucciones en el programa donde utilices esta
  unidad
     Gd:=Detect;
     InitGraph(Gd,Gm,'');
     ResetMouse(ErrCode);     *)

INTERFACE

Uses  Graph,Mouse,CRT,Menu,Dos
Type
   ss      = String[12];
   LengthStr = String;
   TipoHr   = Record
              hr,mi,se,s100 : Word;
            End;
   TipoFe   = Record
              Yr,mo,dy,dw : Word;
            End;

Procedure Aviso;
Procedure Pausa;
Procedure Ventana(Var Opcion:char);
Procedure Zeten(leyenda:lengthStr,Var Uhr:TipoHr;Var Tag:TipoFe;
            Archivo:ss;Escribe:Char);
```

```
IMPLEMENTATION
Var
   m,n,cuadro:Integer;
   ancho,alto,Tam:word;
   I,J,X,Y,But,Gd,Gm,ErrCode:Integer;
   Car:Char;

Procedure AVISO;
Begin
   Sound(600);
   Delay(100);
   Sound(400);
   Delay(100);
   Sound(200);
   Delay(100);
   NoSound;
End;

Procedure PAUSA;
Begin
   While KeyPressed Do
     Car:=ReadKey;
   Repeat until Keypressed;
   While KeyPressed Do
     Car:=ReadKey;
End;

Procedure VENTANA;   {Da el menú de ayuda}
Var
   Op:Integer;

Begin
   SetWriteMode(NormalPut);
   SetMenu('Abortas');
   SetMenu('Otra vez');
   SetMenu('Salva');
   Op:=MenuTo(400,50,'MENU');
   Case Op of
      1: Car:='A';
      2: Car:='O';
      3: Car:='S';
   End;
   CloseBoard;
   Opcion:=Car;
   SetViewPort(0,0,GetMaxX,GetMaxY,True);
   SetCursLim(0,0,GetMaxX,GetMaxY);
End;

Procedure Zeiten;

Begin
   GetTime(Uhr.hr,Uhr.mi,Uhr.se,Uhr.s100);
   GetDate(Tag.Yr,Tag.Mo,Tag.Dy,Tag.Dw);
   Writeln(leyenda);
   Escribe:=Upcase(Escribe);
   Case Escribe of
      'H' : Writeln(archivo,':2,'TIEMPO :  ',Uhr.hr,'h ',Uhr.mi,'m ',Uhr.se,
                '  ',Uhr.s100,'s100');
      'F' : Writeln(archivo,':2,'FECHA (DD:MM:YY) : ',Tag.Dy,'/',Tag.Mo,'/',
                Tag.Yr);
      'A' : Begin
            Writeln(archivo,':2,'FECHA (DD:MM:YY) : ',Tag.Dy,'/',
                Tag.Mo,'/',Tag.Yr);
            Writeln(archivo,':2,'TIEMPO :  ',Uhr.hr,'h ',Uhr.mi,
                'm ',Uhr.se,'  ',Uhr.s100,'s100');
            End;
   End;
End;
```

```
End. {menu}
```

C.V Manejo de la tarjeta DT2851 (Unidad IRIS).

```
{$I HEAD}

UNIT Iris;

(* Esta unidad captura la imagen a partir de VCR o camara de video mediante
  el uso de la tarjeta digitalizadora DT2858 (IRIS) controlada por medio
  de Pascal. *)

INTERFACE

Type
  Id=String[12];

Procedure Captura(identificador:Id;posicion:boolean;Var PosX,PosY:integer);

IMPLEMENTATION
Uses
  Mouse,Extend,CRT,Genoa,Menudo;

Const Base = $310;   { Base address in I/O space }

    INCSR1 = Base + $0;
    INCSR2 = Base + $2;
    OUTCSR = Base + $4;
    CROSS  = Base + $6;
    INDEX  = Base + $8;
    INLUT  = Base + $A;
    REDGRN = Base + $C;
    BLUE   = Base + $E;

Type
  Imagen=string[12];

Var  I,J,k,Lut,
    x,y,xp,yp:Integer;
    Car:Char;
    s:String[1];
    But:Integer;
    Goofy : Array[0..30000] of Byte;
    Tabla:Array[0..256,0..2] of Byte;

PROCEDURE Captura;

Procedure Save(ima:Imagen);

Var  I,ErrCode:Integer;
    Out:File;

Begin
    Assign(Out,ima);
    ReWrite(Out);
    For I:=0 to 3 Do
      Begin
        Locate(I);
        BlockWrite(Out,Mem[$A000:$0000],511);
        BlockWrite(Out,Mem[$A000:$FF80],  1);
      End;
    Close(Out);
End;

Procedure InLUTSel(Indx:Integer);
Begin
  PortW[INCSR1]:=(PortW[INCSR1] and $FFF8) or (Indx and $07);
End;
```

```
Procedure Buffer(Indx:Integer);
Begin
  PortW[INCSR2]:=PortW[INCSR2] and ((Indx and $01) shl 7);
End;

Procedure Display;
Begin
  PortW[OUTCSR]:=PortW[OUTCSR] or $00C0;  {CON $00C0 SE ENCIENDE EL CURSOR}
End;

Procedure Sync(Indx:Integer);
Begin
  PortW[OUTCSR]:=(PortW[OUTCSR] and $FF0F) or ($A0);
End;

Procedure SetBuf(Indx:Integer);
Begin
  PortW[OUTCSR]:=PortW[OUTCSR] and ((Indx and $01) shl 4);
End;

Procedure OutLUTSel(Indx:Integer);
Begin
  PortW[OUTCSR]:=(PortW[OUTCSR] and $F8) or (Indx and $07);
End;

Procedure Go;
Begin
  PortW[INCSR1]:=(PortW[INCSR1] and $F7) or $90;
End;

Procedure Stop;
Begin
  PortW[INCSR1]:=PortW[INCSR1] or $08;
End;

Procedure Mode(Indx:Integer);       {f8 limpia incsr2}
Begin
  PortW[INCSR2]:=(PortW[INCSR2] and $FF8F) or ((Indx and $07) shl 4);
End;

Procedure Cursor(x,y:Integer);
Begin
  PortW[CROSS]:=Word((y shl 8) or x);
End;

Procedure Dup;

Var   I,ErrCode:Integer;

Begin
  For I:=0 to 3 Do       {def 0-3}
    Begin
      Locate(I);
      Transfer($A00000+I*$10000,$A0000,$FFFF,ErrCode);
    End;
End;

Procedure Inicia;
Begin
  PortW[OUTCSR]:=PortW[OUTCSR] or $0000;        {DISPLAY OFF}
  PortW[INCSR1]:=PortW[INCSR1] or $08;          {SET ENSTOP}
  PortW[INCSR1]:=PortW[INCSR1] or $08;          {CLEAR BUSY}
                                {DA INLut TABLE}
                                {DA OUTLut TABLE}
  PortW[INCSR2]:=PortW[INCSR2] and $FF90;       {INICIA INCSR2 BUT LOAD LUT MODE}
                                {INICIA MEMORIA}
  PortW[OUTCSR]:=PortW[OUTCSR] or $0080;        {SELECCIONA DIS ON, CURSOR}
End;
```

```
Begin  {+++++++++   Principal Captura  -  +++++++++}

   Inicia;
   For I:=0 to 63 Do
     For K:=4*I to 4*(I+1) Do
       For J:=0 to 2 Do
         Tabla[K,J]:=I;

   OpenGraph(VGA512);
   SetAllColorRegs(Seg(Tabla),Ofs(Tabla));
   ResetMouse;
   SetCursLim(0,0,255,239);   {default 1000,1000}
   SelBuf(0);
   Mode(1);
   Sync(1);
   OutLUTSel(0);
   InLUTSel(0);
   x:=0;
   y:=0;
   Display;
   Go;
   Lut:=0;
   car:=#0;
   repeat
      Delay(150);
      Dup;{}
   until KeyPressed;
   Aviso;
   repeat
      If not posicion then
      Begin
         GetButtonStatus;
         X:=MouseX;
         Y:=MouseY;
         If (X<>Xp) or (Y<>Yp) Then
           Begin
             Cursor(X,Y);
             Xp:=X;
             Yp:=Y;
           End;
         PosX:=X;  PosY:=Y;    {doy la última posición del cursor}
      End
      Else
         Cursor(PosX,PosY);
   until AnyMouseKeyPressed;
   Stop;
   Dup,{}
   Delay(200);
   Save(identificador);  {}
   Aviso;
End;  {Captura}

END.  {unidad IRIS}
```

C.VI Muestreo de la Imagen (Unidad Scan2).

```
{$I HEAD}

UNIT Scan2;

(* Este program despliega graficos en 16 niveles, sobre esa imagen se busca
   la zona de muestreo mediante circulos y se toman las coordenadas de los
   puntos buscados, entonces se despliega la imagen en 256 niveles leyendo
   el valor de los pixeles de interes, guardandose el resultado en un archivo *)

INTERFACE
```

```
Type
  archi=String[11];

Procedure Scan(muestra:archi;rotacion:Boolean);

IMPLEMENTATION
Uses  Graph,CRT,
    Geno,      {Unidad modificada de Genoa   }
    Mouse,     {Nueva unidad del Mouse de Fer}
    Menudo,    {Utilerja mja                 }
    Menu;

Procedure Scan;
Const
   YAsp  = 1.257;  XAsp  = 1;
   On    = True;   Off   = False;
   Paso  = 30;  {c/30 grados}
   Puntos= 12;  {ptos por circulo}
   circulos = 5;
Type
    palabra = String[15];

Var
    Tabla:Text;    msg:String[20];
    Red,Green,Blue,
    wide           :Integer;
    Dat            :String[11];
    Respuesta,Eleccion :Char;
    _F      :Real;
    archivo  :File of Byte;
    Poly    :Array[0..8,1..2] of Integer;
    Pixel   :array[1..5,1..12] of Byte;      {# circulos, ptos/circ}
    Buffer  :Array[0..512]    of Byte;
    PixTemp  :Word;
    Tam,TamA,TamB,TamC,TamD  :word;
    Zona3,Zona3a,Zona3b,
    Zona3c,Zona3d              :Pointer;
    I,J,K,L,X,Y,But,Gd,Gm,
    ErrCode,ratio,alt,ancho,
    ancho1,xpix,ypix,radio   :Integer;
    tono     :Boolean;
    Inter    :File;
    Car      :Char;
    rad,rad2,leta :real;
    xm,ym,ax,ay,X1,
    Y1,Xo,Yo,
    n1,n2:longint;

Procedure Despliega(imagen:palabra;radii:longint;
        rt:Boolean;gris:Boolean);
{   Vacia la imagen en pantalla en 256 niveles y les los puntos      }
{   seleccionados con la tarjeta Genos                               }
Var
  I,J,z1,z2:Integer;      {numero de circulos que se muestrean}

Begin
    Assign(Inter,Imagen);       {Despliegue de la imagen}
    Reset(Inter);
    If IOResult<>0 Then
        Halt(1);
    OpenGraph;    { abre graficas 512x512x256 de geno }
    For I:=0 to 3 Do
      Begin
        Locate(I);
        BlockRead(Inter,Mem[$A000 $0000],511);
        BlockRead(Inter,Mem[$A000:$FF7F],1);
      End;

    If gris then
```

```
      Begin
        For I:=0 to 63 Do          {Esta tabla es en niveles de gris}
          For J:=4*I to 4*(I+1)-1 Do
            SetColorReg(J,I,I,I);
      End
    Else
      Begin
        Assign(Tabla,'Color.');        {Esta tabla esta en colores}
        Reset(Tabla);
        For I:=0 to 255 Do
          Begin
            ReadLn(Tabla,Red,Green,Blue);
            SetColorReg(I,Red,Green,Blue);
          End;
        Close(Tabla);
      End;

  {Lee los puntos de los circulos seleccionados}
  K:=1;
  For I:=1 to circulos Do
    Begin
      rad:=radii*I/circulos;
      L:=0;
      if rt then J:=180
      Else J:=0;
      Repeat
        teta:=J*2*pi/360;
        xpix:=Round(xm+rad*cos(teta));
        ypix:=Round(ym+rad*sin(teta));
        Inc(J,Paso);
        Inc(L);          {PixelG es de la unidad geno.pas}
        PixTemp:=0;
        For z1:=3 Downto 1 Do
          For z2:=3 Downto 1 Do   {lee 9 pixeles en periferia}
            PixTemp:=PixTemp+PixelG(xpix+z1-2,ypix+z2-2);
        Pixel[K,L]:=PixTemp div 9;
      until L=puntos;                {promediando 9 ptos}
        {promedia los 9 puntos}
      Inc(K);
    End;
  Close(Inter);
End;  {Despliega}

Procedure Pinta(semidia,PosY:Integer);
Begin
  SetWriteMode(XORPut);        {pinta negro para colocar TAM}
  SetFillStyle(SolidFill,Black);
  wide:=TextWidth('1234');
  Alt:=TextHeight('L');
  Bar(520+((119-wide)div 2),PosY+((17-alt)div 2),
     630,PosY+18 );
  SetTextStyle(DefaultFont,HorizDir,1);
  wide:=TextWidth(Trans(Semidia,_Integer));
  OutTextXY(520+((121-wide)div 2),PosY+((17-alt)div 2),
     Trans(Semidia,_Integer));    {coloca TAM}
End;

Procedure GetCircle(cx,cy:word;Var R:integer);
{+ Este procedimiento dibuja un poligono para despues dar un circulo con
 fines de determinaci¢n del rea a muestrear                    ++}
Var
  Radios  :Integer;

Procedure CalcCirc;
Var
  Factor  : Integer;

Begin
  Factor:=Round( _F*R);
```

```
  Poly[0,1]:=X+R;     Poly[0,2]:=Y+Factor;
  Poly[1,1]:=X+R;     Poly[1,2]:=Y-Factor;
  Poly[2,1]:=X+Factor; Poly[2,2]:=Y-R;
  Poly[3,1]:=X-Factor; Poly[3,2]:=Y-R;
  Poly[4,1]:=X-R;     Poly[4,2]:=Y-Factor;
  Poly[5,1]:=X-R;     Poly[5,2]:=Y+Factor;
  Poly[6,1]:=X-Factor; Poly[6,2]:=Y+R;
  Poly[7,1]:=X+Factor; Poly[7,2]:=Y+R;
  Poly[8,1]:=X+R;     Poly[8,2]:=Y+Factor;

End;

Begin
 X:=cx;
 Y:=cy;
 _F:=Sqrt(2.0)-1.0;
 SetWriteMode(XorPut);   {estaba en XORPut}
 ShowMouseCursor;
 R:=0;
 Repeat
   Radios:=Abs(Integer(MouseX-X));
   If Radios<>R Then
     Begin
       HideMouseCursor;
       CalcCirc;
       DrawPoly(9,Poly);
       Pinta(Radios,438);
       R:=Radios;
       CalcCirc;
       DrawPoly(9,Poly);
       Pinta(Radios,438);
       ShowMouseCursor;
     End;
 Until AnyMouseKeyPressed;
 HideMouseCursor;
 DrawPoly(9,Poly);
 ShowMouseCursor;
End;

Procedure Sonido;
Begin
   Sound(500);
   Delay(100);
   NoSound;
End;

Function Recorta(ss:String) : String;
Var P: Integer;
Begin
   P:=Pos(',',ss);
   Recorta:=Copy(ss,1,P-1);
End;
(*   ***************   PROGRAMA   PRINCIPAL   ******************)

BEGIN     {Scan}
   {con 16 tonos, define el area de muestreo}
   Gd:=Detect;
   InitGraph(Gd,Gm,'');
   ResetMouse;
   SetCursLim(0,0,512,GetMaxY);
   Dat:=Recorta(muestra);
   Assign(Inter,muestra);
   Reset(Inter);
   For J:=0 to $F Do
     Begin
       SetPalette(J,J);
       SetRGBPalette(J,4*J,4*J,4*J);
     End;
```

```
{++++++    Despliega la PALETTE colocada lateralmente          ++++++++++}
{++++++    y cual es la imagen desplegada en un aviso          ++++++++++}

  For I:=0 to $F Do
    Begin
      SetFillStyle(SolidFill,$F-I);
      Bar(600,I shl 4,639,(I+1) shl 4);
    End;
  SetWriteMode(XORPut);
  Rectangle(520,260,634,260+40);
  Rectangle(522,262,632,260+38);
  SetFillStyle(WideDotFill,12);
  Bar(523,263,631,277);
  Line(523,277,631,277);
  SetTextStyle(DefaultFont,HorizDir,1);
  alt:=TextHeight('Hola');          {ponerla en la esq inf der}
  ancho:=TextWidth('IMAGEN:');
  OutTextXY(520+((119-ancho)div 2),264+((17-alt)div 2),'IMAGEN:');
  ancho1:=TextWidth('Prufa2e.img');
  OutTextXY(520+((119-ancho1)div 2),262+((17-alt)div 2),Muestra);
  Rectangle(520,420,634,420+40);     {CUADRO P RADIO}
  Rectangle(522,422,632,420+38);
  SetFillStyle(WideDotFill,12);
  Bar(523,423,631,437);
  Line(523,437,631,437);
  SetTextStyle(DefaultFont,HorizDir,1);
  alt:=TextHeight('H');          {ponerla en la esq inf der}
  ancho:=TextWidth('RADIO:');
  OutTextXY(520+((119-ancho)div 2),424+((17-alt)div 2),'RADIO:');
{++++++++++++     Despliegue de la imagen          +++++++++++++++}
  Reset(Inter);
  For J:=0 to 511 Do
    Begin
      BlockRead(Inter,Buffer,4);
      For I:=0 to 511 Do
        PutPixel(I,J,Buffer[I] shr 4);
    End;
  Close(Inter);

{ La deformaci¶n del despliegue de la tarjeta Gr fica se da aqui    }
  SetAspectRatio(Trunc(YAsp*10000),Trunc(XAsp*10000));     {}
  {Delimitaci¶n de la Zona}
  Repeat                {LOOP para muestrear}
    ShowMouseCursor;          {y donde hay que borrarlo     }
    Sonido;
    Repeat
      GetButtonStatus;
    Until AnyMouseKeyPressed;
    Xo:=MouseX;
    Yo:=MouseY;
    SetColor(Yellow);
    Sonido;
    GetCircle(Xo,Yo,radio);          {Determina de cuanto es el  rea}
    xm:=Xo;
    ym:=Yo;
    HideMouseCursor;
    {Aqui se guarda la zona donde se dibujan los circulos}
    {Y parte la imagen en 4 si rebasa los 64k-$f}
    rad2:=YAsp*radio;
    Tam:=ImageSize(xm-Round(Round(rad2))-4,ym+Round(rad2)+4,
              xm+Round(rad2)+4,ym-Round(rad2)-4);
    If (Tam>65518) Or (Tam=0) then
      Begin
        TamA:=ImageSize(xm-Round(rad2)-4,
                  ym-Round(rad2)-4,
                  xm,ym);
        GetMem(Zona3a,TamA);
        GetImage(xm-Round(rad2)-4,ym-Round(rad2)-4,
```

```
            xm,ym,Zona3a^);
        TamB:=ImageSize(xm,ym-Round(rad2)-4,
                xm+Round(rad2)+4,ym);
        GetMem(Zona3b,TamB);
        GetImage(xm,ym-Round(rad2)-4,
                xm+Round(rad2)+4,ym,Zona3b^);
        TamC:=ImageSize(xm-Round(rad2)-4,ym,
                xm,ym+Round(rad2)+4);
        GetMem(Zona3c,TamC);
        GetImage(xm-Round(rad2)-4,ym,
                xm,ym+Round(rad2)+4,Zona3c^);
        TamD:=ImageSize(xm,ym,
                xm+Round(rad2)+4,ym+Round(rad2)+4);
        GetMem(Zona3d,TamD);
        GetImage(xm,ym,
                xm+Round(rad2)+4,
                ym+Round(rad2)+4,Zona3d^);
      End
    Else
      Begin
        GetMem(Zona3,Tam);
        GetImage(xm-Round(rad2)-4,ym+Round(rad2)+4,
                xm+Round(rad2)+4,ym-Round(rad2)-4,
                Zona3^);
      End;
                    {Dibuja la zona escogida}
    SetLineStyle(SolidLn,0,1);
    SetColor(yellow);
    If Radio*YAsp<GetMaxY/2 then
      Circle(xm,ym,round(radio));
    Aviso;
    Ventana(Eleccion);
    Car:=Eleccion;
    Car:=UpCase(Car);
    If Car='O' Then    {#79 = 'O'}
      Begin
        If (Tam>65518) Or (Tam=0) then
          Begin

            PutImage(xm-Round(rad2)-4,
                ym-Round(rad2)-4,
                Zona3a^,NormalPut);
            FreeMem(Zona3a,TamA);
            PutImage(xm,ym-Round(rad2)-4,
                Zona3b^,NormalPut);
            FreeMem(Zona3b,TamB);
            PutImage(xm-Round(rad2)-4,ym,
                Zona3c^,NormalPut);
            FreeMem(Zona3c,TamC);
            PutImage(xm,ym,
                Zona3d^,NormalPut);
            FreeMem(Zona3d,TamD);
          End
        Else
          Begin
            PutImage(xm-Round(rad2)-4,
                ym-Round(rad2)-4,Zona3^,NormalPut);
            FreeMem(Zona3,Tam);
          End;
      End;
    SetWriteMode(XORPut);        {pinta negro para colocar TAM}
    SetFillStyle(SolidFill,Black);
    BAr(520+((119-ancho1)div 2),320+((17-alt)div 2),
      GetMaxX-1,320+20 );
  Until (Car='A') or (Car='S');      {65-A,79-O,83-S,97-a,98-b}
  Aviso;
{+++ con 256 niveles, lee la intensidad del pixel guarda los 360 puntos ++++++}
{******   Escribe los datos muestreados en un archivo WORD          *******}
{******   utiliza la tarjeta gr fica GENDA                          *******}
```

```
    If Car='S'then
      Begin
        CloseGraph;
        TextMode(CO80);
        tono:=Off;
        Despliega(muestra,radio,rotacion,tono);  {Usa GENOA}
        Pausa;
        If Dat<>'' then
          Begin
             Aviso;
             if Rotacion then Assign(archivo,Dat+'.IDT')
             Else Assign(archivo,Dat+'.EDT');
             Rewrite(archivo);
             For I:=1 to circulos Do
                For J:=1 to Puntos Do
                   Write(archivo,Pixel[I,J]);
             Close(Archivo);
          End;
        TextMode(CO80);
      End;
      InitGraph(Gd,Gm,'');
    End;
END.  {muestreo}
```

C.VII Diferencia de Intensidad del un par de Imágenes (Unidad Mrdd3).

```
{$I HEAD}

UNIT MRDD3;
{ Tercera generación                                        }
{ Este programa realiza la lectura de los puntos seleccionados por la unidad}
{ MUESTREA (360) de dos imagenes y le aplica el algoritmo de Roddier        }
{ Perteneciente a:  Salvador Fierro Pacheco                 }
{ IAUNAM Diciembre 90'                                      }
INTERFACE
Uses Tipo;
Type
   Wdata = File of Byte;
   Sdata = File of Single;
   nom   = String[12];
Var
  sust1,sust2  :Wdata;
  resu         :Sdata;

Procedure CTE(NOMBRE:String);  {nombre sin extension}

Procedure Metodo(Dat : nom; Var Sol,Scoef : RealArrayMA;
          Ajuste : Integer);        {On 10 var, Off 4 var}
  { Escribirlo con MAYUSCULAS los parametros}

IMPLEMENTATION
Uses DOS,Graph,Crt,
    FoolGraf,Menu,Menudo,Mouse,LSFit;  {Linear Square Fit}

PROCEDURE CTE;
Var
  Ctes        : Array[1..4] of Real;
  i           :Integer;
  lamda,f,l,R,ctte : Real;
  archivo     : File of Real;
  Leyenda     :String;
Begin
    For i:=1 to 4 Do
    Begin
       Ctes[1]:=0.6328e-3; {lamda}
       Ctes[2]:=2000;      { f }
       Ctes[3]:=80,        { l }
```

```
      Ctes[4]:=76;      { R }
    End;
    DrawBoard(170,220,525,270,'CONSTANTES :');
    MoveTo(8,4);
    OutText('Ingreso de constantes en el sig. orden');
    MoveTo(8,20);
    OutText('Lamda, f, I, R.');
    MoveTo(8,36);
    SetTextStyle(SmallFont,0,5);
    OutText(' Pinche cualquier tecla para continuar');
    Repeat until KeyPressed;
    Leyenda:=nombre+'.CTE'+' (mm)';
    EditFLVals(120,120,Ctes,4,20,Leyenda);
    CloseBoard;
    HideMouseCursor;
    Ctte:=2*Ctes[1]*Sqr(Ctes[2])/(Ctes[3]*Sqr(Ctes[4]));
    Assign(archivo,nombre+'.CTE');
    Rewrite(archivo);
    Write(archivo,CTTE);
    Close(archivo);
End;

Procedure Metodo;

TYPE
    Datos   = RealArrayMAbyMA;
VAR
  b1,b2            : Array[1..ndatap] of Byte;
  pos              : File of Real;
  d,npar,j,j,k,l,
  NumData          : Integer;
  Dat1,Dat2        : Nom;
  PosR,PosT,PR,PT,
  den,num,cociente  : RealArrayNDATA;
  Mcoef            : RealArrayMA;
  Listafit         : IntegerArrayMFIT;□
  numfit           : Integer;
  MatCov           : RealArrayMAbyMA;
  Tempo,
  chisqr,constante  : Real;
  sust1,sust2      : File of Byte;
  Ct               : File of Real;

BEGIN    {************  Programa  Principal    *************}
   Dat1:=Dat+'.IDT';
   Dat2:=Dat+'.EDT';
   assign(sust1,Dat1);
   assign(sust2,Dat2);
   reset(sust1);
   reset(sust2);

{***********      Calculo el valor de b1-b2/b1+b2 en reales   **********}
   For I:=1 to ndatap Do
     Begin
        Read(sust1,b1[i]);
        Read(sust2,b2[i]);
     End;
   FOR I:=1 TO ndatap DO
     BEGIN
        den[i]:=b1[i]+b2[i];
        num[i]:=b1[i]-b2[i];
        IF ((num[i]=0) OR (den[i]=0)) THEN
          cociente[i]:=0.0
        Else
          cociente[i]:=num[i]/den[i];
     END;  {for}
   CLOSE(sust1);
   CLOSE(sust2);
{******   Aqui se generan los coeficientes para la matriz de ajuste ******}
```

```
      Assign(pos,'POS5.');   {lee la posicion r,theta}
      Reset(pos);
      For I:=1 to ndatap Do
        Begin
          Read(pos,Tempo);
          PR[I]:=Tempo;
          Read(pos,Tempo);
          PT[I]:=Tempo;
        End;
 { ***********************************          Read(pos,PosR[I],PosT[I]); }
    If Ajuste=1 then
      Begin
          numfit:=6;  {num de var a ajustar}
          Listafit[1]:=1;   Listafit[6]:=10;
          Listafit[2]:=2;   Listafit[7]:=0;
          Listafit[3]:=4;   Listafit[8]:=0;
          Listafit[4]:=5;   Listafit[9]:=0;
          Listafit[5]:=9;  Listafit[10]:=0;
          NumData:= 12;
          For I:=1 to numData Do
            Begin
              PosR[I]:=PR[48+I];
              PosT[I]:=PT[48+I];
            End;
      End
    Else
    Begin
        numfit:=4;  {num de var a ajustar}
        Listafit[1]:=3;   Listafit[6]:=0;
        Listafit[2]:=6;   Listafit[7]:=0;
        Listafit[3]:=7;   Listafit[8]:=0;
        Listafit[4]:=8;   Listafit[9]:=0;
        Listafit[5]:=0;  Listafit[10]:=0;
        NumData:=48;    {4 circ x 12 puntos}
        For I:=1 to NumData Do
          Begin
            PosR[I]:=PR[I];
            PosT[I]:=PT[I];
          End;
    End;
    For I:=1 to map Do   {inicia Mcoef}
          Begin
        Mcoef[I]:=0;
            SCoef[I]:=0;
      End;
    ChiSqr:=0.0;

    LFit(PosR,PosT,cociente,
        NumData,
        Mcoef,
        map,                {11}
        ListaFit,numfit,
        MatCov,
        ChiSqr,
        Scoef);     {varianza d Mcoef}
    Sol:=Mcoef;             {pasar como parametro la sigma de c/u}
     Assign(Ct,dat+'.CTE');
     Reset(Ct);
    Read(Ct,constante);
     Close(Ct);
    For I:=1 to map Do
        Sol[I]:=Sol[I]/constante;
   End;
'END.
```

C.VIII Despliegue de la Imagen. (Unidad Disp).

ESTA TESIS NO DEBE
SALIR DE LA BIBLIOTECA


```
{$I HEAD}

Unit Disp;
{ Esta unidad despliega en falso color, RGB y tonos de gris una imagen  }
INTERFACE
Type
   Parametro = String[20];

Procedure SHOW(Filename:Parametro);

IMPLEMENTATION
Uses  Graph,CRT,Menudo,menu,Mouse;
Procedure SHOW;

Var   TabName : String[20];
      Gd,Gm,
      I,ErrCode,alt,ancho,ancho1,
      X,Y,But,
      R,G,B    :Integer;
      Data     :Byte;
      Inter    :File;
      Tabla    :Text;
      Buffer   :Array[0..15,0..511] of Byte;

Const Param = 511;

Begin
  FileName:=FileName;
  Assign(Inter,FileName);
  Assign(Tabla,'COL');
  Reset(Inter,512);
  Reset(Tabla);
  If IOResult<>0 Then
    Begin
      WriteLn('No mames...');
      Repeat Until KeyPressed;
      Halt(1);
    End;
  BlockRead(Inter,Buffer,1);
  Gd:=Detect;
  InitGraph(Gd,Gm,'');

{******    Despliega la PALETTE colocada lateralmente          ***********}
{******    y cual es la imagen desplegada en un marco          ***********}

  For I:=0 to $F Do
    Begin
      SetFillStyle(SolidFill,$F-I);
      Bar(600,I shl 4,639,(I+1) shl 4);
    End;
  SetWriteMode(XORPut);
  Rectangle(520,260,634,260+40);
  Rectangle(522,262,632,260+38);
  SetFillStyle(WideDotFill,12);
  Bar(523,263,631,277);
  Line(523,277,631,277);
  SetTextStyle(DefaultFont,HorizDir,1);
  alt:=TextHeight('Hola');
  ancho:=TextWidth('IMAGEN:');
  OutTextXY(520+((119-ancho)div 2),264+((17-alt)div 2),'IMAGEN:');
  ancho1:=TextWidth('Prufa2e.img');
  OutTextXY(520+((119-ancho1)div 2),282+((17-alt)div 2),FileName);
(*  mensaje para continuar      *)
  Rectangle(514,GetMaxY-53,639,GetMaxY-21);
  Rectangle(514,GetMaxY-51,639,GetMaxY-23);
  SetFillStyle(WideDotFill,7);
  Bar(515,GetMaxY-50,638,GetMaxY-24);
  OutTextXY(518,GetMaxY-46,'Pinche el Mouse');
```

```
  OutTextXY(518,GetMaxY-37,'para continuar!');
  For I:=0 to $F Do
    Begin
      SetPalette(I,I);
      ReadLn(Tabla,R,G,B);
      SetRGBPalette(I,R,G,B);
    End;
  For I:=0 to 29 Do
    Begin
      BlockRead(Inter,Buffer,16);
      For y:=0 to $F Do
        For x:=0 to Param Do
          PutPixel(x,y+16*I,Buffer[y,x] shr 4);
    End;
  Aviso;                          {256 Colores}
  Repeat Until AnyMouseKeyPressed;
  Delay(2000);
  For I:=0 to $F Do
    SetRGBPalette(I,4*I,0,0);          {R,G,B}
  Delay(2000);
  For I:=0 to $F Do
    SetRGBPalette(I,0,4*I,0);
  Delay(2000);
  For I:=0 to $F Do
    SetRGBPalette(I,0,0,4*I);
  Delay(2000);
  For I:=0 to $F Do                  {Gris}
    SetRGBPalette(I,4*I,4*I,4*I);
  Aviso;
  Repeat Until AnyMouseKeyPressed;
  End;
End.
```

C.IX Cálculo de tiempo de procesamiento (Unidad Zeit).

```
{$I HEAD}

UNIT ZEIT;
(* Esta unidad te da el tiempo transcurrido en un intervalo de tiempo
   Enero 91  Courtesy of Abel   *)

INTERFACE

Procedure ClockOn;
Procedure ClockOff(Var Tiempo:Real);
Function DosClock : real;

IMPLEMENTATION
Uses Dos;
VAR
  ClockInicio : Real;
Procedure ClockOn;
Begin
  ClockInicio:=Dosclock;
end;

Procedure ClockOff;
Begin
  writeln('Tiempo Transcurrido : ',(DosClock-ClockInicio):0:3,' segundos');
  Tiempo:=(DosClock-ClockInicio);
end;

Function DosClock : real;
Var  Regs : Registers;
Begin
  Regs.AH := $2C;
  MsDos(Regs);
  With Regs do
```

```
  DosClock := CH*3600.0 + CL*60.0 + DH + DL/100.0;
end;

END.
```

C.X Encabezado común de configuración (Unidad HEAD).

```
{$A+,B-,D+,E+,F-,G+,I+,L+,N-,O-,R+,S+,V-,X-}
{$M 16384,0,655360}
```

APENDICE D

Cálculo de aberraciones del divisor de cubo.

A continuación se presenta el cálculo de las aberraciones del divisor de cubo y la razón por la cual aberración esférica y astigmatismo se consideran despreciables, despreciándose en el análisis de aberraciones para la prueba de Roddier con un haz abierto a $f/12$.

Se consideran aquí , el divisor de cubo como un plano de vidrio de caras planas y paralelas. Las aberraciones están dadas por Welford[15]

D.I Aberración esférica

$$S_I = -\frac{1}{8}\frac{h^4}{x^4}\frac{n^2-1}{n^3}d$$

si $d = 25$ mm y $\frac{h}{x} = \frac{1}{2fl}$

$$\begin{aligned} S_I &= -\frac{1}{8}\left(\frac{1}{2fl}\right)\frac{(1.5)^2-1}{(1.5)^3}(25) \\ &= -\frac{1}{8}\left(\frac{1}{24}\right)^4\frac{2.25-1}{3.37}(25) \\ &= 3.49\times 10^{-6}\text{mm} = 0.003\mu\text{m} \\ &\Rightarrow \frac{\lambda}{167} \qquad (\text{para}\lambda = 0.5\mu\text{m}) \end{aligned}$$

D.II Astigmatismo

$$S_{III} = -\frac{1}{4}\frac{h^2}{x^2}\frac{k^2}{p^2}\frac{n^2-1}{n^3}d$$

$\frac{h^2}{x^2}$ es pequeño pero p es muy grande entonces $p = 1000$ y k no rebasa el valor de $k = 12.5$ Calculando S_{III}

$$\begin{aligned} S_{III} &= -\frac{1}{4}\left(\frac{1}{24}\right)^2\left(\frac{12.5}{1000}\right)^2\frac{2.25-1}{3.37}(25) \\ &= 629\times 10^{-9}\text{mm} = 0.0006\mu\text{m} \\ &\Rightarrow \frac{\lambda}{1000} \end{aligned}$$

$$S_{IV} = 3S_{III}$$

De aquí que las aberraciones sean despreciables.

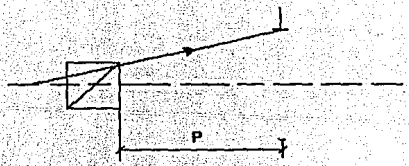

Figura D.I. Divisor de cubo

Referencias

[1] E. Ruiz, S. Cuevas, et al., "Offset Guider/Alignment System for the 2.1m UNAM Telescope", SPIE **627**, 231-235 (1986).

[2] J.M. Beckers, J.T. Williams, MMT Technical Memorandum 79-1, Oct. (1979).

[3] F. Roddier, "Curvature Sensing and Compensation: a New Concept in Adaptive Optics", Appl. Opt., **27**, 1233-1235 (1988).

[4] A. Behr, "A Proposal for the Alignment of Large Telescopes", Astron. Astrophys., **28**, 355-358 (1973).

[5] C. Roddier, F. Roddier, "Testing of Telescope Optics: a new approach", Proc. of SPIE,**1236**, 756-766 (1990).

[6] K. Ichikawa, A.W. Lohmann, M. Takeda, "Phase Retrieval Based on the Irradiance Transport Equation an the Fourier Transform Method: Experiments", Appl. Opt., **27**, 3433-3436 (1988).

[7] Goodman, "Introduction to Fourier Optics", McGraw-Hill (1968).

[8] R. Noll, "Zernike Polynomials and Atmospheric Turbulence", J. Opt. Soc. Am., **66**, 207-211 (1976).

[9] Data Translation User Manual for DT2851 High resolution frame grabber for the IBM Personal Computer AT, (1988).

[10] D. Malacara, "Optical Shop Testing", Wiley, 2nd. edit. cap. 14 (1992).

[11] C. Roddier, F. Roddier, "Wavefront reconstruction from defocused images and the testing of ground-based optical telescopes", submitted to J. Opt Soc. Am. A (1993).

[12] M. Born, E. Wolf, Principles of Optics, sec. 9.2, Pergamon Press (1965).

[13] D.L. Fried, J. Opt. Soc. Am., **55**, 1427 (1965).

[14] W. Press et all, Numerical Recipes in Pascal (The Art of Scientific Computing), Press, Cambrigde University Press, (1989).

[15] W.T. Welford, "Aberrations of symmetrical optical system", Academic Press ed., 229 (1974).