

CIMAT

Centro de Investigación en Matemáticas, A.C

Estrategias Evolutivas Aplicadas al Problema de Registro de Imágenes

Tesis

que para obtener el grado de

**Maestro en Ciencias con Especialidad en Computación y
Matemáticas Industriales**

presenta

Héctor Fernando Gómez García

Asesores

Dr. José Luis Marroquín Zaleta. CIMAT.

Dr. Arturo Hernández Aguirre. CIMAT.

Dr. Salvador Botello Rionda. CIMAT.

Guanajuato, Gto. Agosto de 2002.

Coxaca
20164

Índice General

| | | |
|----------|---|-----------|
| 1 | Introducción | 3 |
| 1.1 | Objetivos de la Tesis | 4 |
| 1.2 | Descripción de la Estructura de la Tesis | 4 |
| 2 | Registro de Imágenes | 7 |
| 2.1 | Planteamiento General del Problema de Registro. | 7 |
| 2.2 | Tipos de Transformaciones | 8 |
| 2.2.1 | Transformaciones Paramétricas | 8 |
| 2.2.2 | Transformaciones no Paramétricas | 10 |
| 2.3 | Modalidades del Registro de Imágenes | 10 |
| 2.3.1 | Registro Monomodal | 10 |
| 2.3.2 | Registro Multimodal | 11 |
| 2.4 | Esquema Multiescala | 11 |
| 3 | Estrategias Evolutivas | 13 |
| 3.1 | Antecedentes | 13 |
| 3.2 | Descripción de las Estrategias Evolutivas. | 14 |
| 3.2.1 | Codificación | 15 |
| 3.2.2 | Función de Aptitud | 16 |
| 3.2.3 | Recombinación | 16 |
| 3.2.4 | Mutación | 17 |

| | | |
|----------|---|-----------|
| 3.2.5 | Selección | 19 |
| 3.2.6 | Algoritmo General | 21 |
| 4 | Registro de Imágenes Basado en Intensidad | 25 |
| 4.1 | Suma de Diferencias al Cuadrado | 26 |
| 4.1.1 | Método de Gauss-Newton | 26 |
| 4.2 | Estrategias Evolutivas Aplicadas a Registro Basado en Intensidad | 31 |
| 4.2.1 | Codificación | 32 |
| 4.2.2 | Función de Aptitud | 32 |
| 4.2.3 | Recombinación, Mutación y Selección | 33 |
| 4.2.4 | Esquema Multiescala | 33 |
| 4.3 | Comparación entre Algoritmos de Gauss-Newton y Estrategias Evolutivas Aplicados a Registro Basado en Intensidad | 34 |
| 4.3.1 | Pruebas en Imágenes no Ruidosas | 36 |
| 4.3.2 | Pruebas en Imágenes Ruidosas | 39 |
| 4.3.3 | Conclusiones sobre las pruebas | 39 |
| 5 | Registro de Imágenes Basado en Información Mutua | 41 |
| 5.1 | Antecedentes | 41 |
| 5.1.1 | Ventanas de Parzen | 43 |
| 5.2 | Registro por Maximización Estocástica de la Información Mutua | 44 |
| 5.3 | Registro por Maximización de la Información Mutua mediante Estrategias Evolutivas | 47 |
| 5.4 | Comparación entre Estrategias Evolutivas y Ascenso de Gradiente Estocástico | 48 |
| 6 | Conclusiones y Trabajo Futuro | 51 |

Capítulo 1

Introducción

La Computación Evolutiva está formada por un conjunto de técnicas de optimización que imitan el proceso evolutivo de las especies en la naturaleza. Entre estas técnicas encontramos a los Algoritmos Genéticos, la Programación Evolutiva y las Estrategias Evolutivas. Curiosamente todas tienen sus inicios en la década de los 60's [7],[8].

John Holland [1] introdujo los algoritmos genéticos en los Estados Unidos buscando modelar el proceso de adaptación de las especies, posteriormente Kenneth de Jong [2] los utilizó en problemas de optimización paramétrica. La programación evolutiva fue introducida por Lawrence J. Fogel [3] en los Estados Unidos y desarrollada hasta su estado actual por D. Fogel [4]. Finalmente las estrategias evolutivas fueron utilizadas originalmente por Ingo Rechenberg [5] en problemas de optimización entera y posteriormente desarrollada por Paul Schwefel [6], ambos en Alemania.

Estos algoritmos se basan en poblaciones de individuos que representan puntos de búsqueda sobre el espacio de parámetros a optimizar, los cuales evolucionan a través de procesos aleatorios de recombinación, mutación y selección de manera que cada vez que se aplican tales procesos, los individuos se distribuyen en mejores regiones.

La computación evolutiva ha sido aplicada con gran éxito a problemas de optimización en los cuales las técnicas tradicionales no trabajan satisfactoriamente, siendo considerada

como una herramienta de optimización muy prometedora, sin embargo su desempeño difícilmente es ventajoso en problemas en que las técnicas tradicionales son aplicadas con éxito.

El Registro de Imágenes es una técnica que pertenece al Procesamiento Digital de Imágenes y que se destaca por la importancia de sus aplicaciones. Esta técnica requiere la utilización de métodos de optimización y su éxito depende totalmente del desempeño de tales métodos. En ese sentido, se han realizado esfuerzos para aplicar la computación evolutiva al registro de imágenes, [9] y [10], han utilizado los algoritmos genéticos para resolver este problema..

En este trabajo se explora la aplicación de las estrategias evolutivas al registro de imágenes.

1.1 Objetivos de la Tesis

Los objetivos del presente trabajo se resumen a continuación

- 1.- Proponer algoritmos de registro de imágenes basados en la utilización de estrategias evolutivas como técnicas de optimización.
- 2.- Que los algoritmos propuestos sean robustos al ruido y que se desempeñen con éxito ante transformaciones grandes.
- 3.- Realizar una comparación entre los algoritmos de registro de imágenes tradicionales y los propuestos.

1.2 Descripción de la Estructura de la Tesis

Este trabajo se ha desarrollado en seis capítulos.

En el segundo capítulo se plantea, se describe y clasifica el problema de registro de imágenes.

En el tercer capítulo se revisa el desarrollo histórico de las estrategias evolutivas, se definen formalmente y se describen de manera detallada.

El registro de imágenes basado en comparación de intensidades se estudia en el capítulo cuarto. Se describe el método de Gauss-Newton aplicado a la Suma de Diferencias al Cuadrado, se describe una metodología basada en estrategias evolutivas y al final se comparan ambos enfoques.

El quinto capítulo trata el registro de imágenes basado en la maximización de la información mutua. Se describe una metodología basada en ascenso de gradiente estocástico, se propone un algoritmo basado en estrategias evolutivas y se comparan ambos.

Finalmente en el sexto capítulo se presentan las conclusiones sobre esta tesis y se esbozan trabajos que han quedado pendientes y que sería interesante revisar en el futuro próximo.

Capítulo 2

Registro de Imágenes

2.1 Planteamiento General del Problema de Registro.

El registro es una técnica de gran importancia dentro del procesamiento digital de imágenes. Es utilizado para alinear espacialmente una o más imágenes adquiridas generalmente en instantes de tiempo diferentes, desde perspectivas distintas, o mediante varias modalidades. Su aplicación ha resultado una herramienta de gran utilidad en varias áreas, entre las que destacan la visión computacional, el procesamiento de datos adquiridos remotamente y la medicina, en esta última es utilizada con fines de diagnóstico de enfermedades principalmente [11].

Matemáticamente afirmamos que un par de imágenes I_1, I_2 se encuentran registradas, si podemos encontrar una transformación $T(x)$, tal que la siguiente igualdad se cumpla

$$\phi\{I_1[T(x)]\} = I_2(x) \quad (2.1)$$

A lo largo de este trabajo, llamaremos a I_1 *imagen de referencia*, I_2 *imagen objetivo*, $T(x)$ es una *transformación de coordenadas* que toma su argumento x de Z^2 en el caso de

imágenes bidimensionales, o de Z^3 para arreglos de imágenes o imágenes tridimensionales, $\phi(I)$, es una *función de transferencia de tonos* que actúa relacionando las intensidades en cada par de imágenes.

El registro de imágenes puede ser clasificado de acuerdo a la naturaleza de los operadores $T(x)$ y $\phi(I)$, entre otros factores [12], como se describe en las siguientes secciones.

2.2 Tipos de Transformaciones

El término $I_1 [T(x)]$ representa la imagen obtenida al aplicar la transformación $T(x)$ a la imagen de referencia. Como se había comentado anteriormente el argumento de dicha transformación son las coordenadas enteras de la imagen $I_1 [T(x)]$ y transforma a coordenadas reales de I_1 por lo que debe utilizarse interpolación numérica para evaluar el valor adecuado y así construir la imagen $I_1 [T(x)]$.

En la presente sección se analizan los tipos de transformaciones más importantes en el registro de imágenes.

2.2.1 Transformaciones Paramétricas

Cuando la transformación de coordenadas puede ser definida por medio de un conjunto pequeño de parámetros $\theta = \{\theta_1, \dots, \theta_m\}$, la transformación se denomina *transformación paramétrica*. En ese caso podemos escribir $T(\theta, x)$.

Transformación Rígida

Una transformación de coordenadas se denomina *rígida* si está compuesta de rotaciones y traslaciones únicamente [16]. Una transformación rígida no cambia ni las distancias entre puntos ni los ángulos entre rectas, después de aplicada.

Puede expresarse de la siguiente forma

$$T(\theta, x) = Rx + t$$

R , es una matriz de rotación general y t es un vector de traslación.

El vector de parámetros θ , estará compuesto por los ángulos de rotación de R con respecto a algunos ejes coordenados, y por las componentes del vector de traslación t . En 2D una transformación rígida es parametrizada por tres valores, un ángulo de rotación y dos componentes del vector de traslación, mientras que en 3D, se precisan 6 parámetros.

Transformación Afín

Una *transformación afín* tiene la propiedad de conservar el paralelismo entre rectas una vez aplicada. Dicha transformación puede expresarse como la multiplicación de una matriz general A por las coordenadas x , más un vector de traslación t

$$T(\theta, x) = Ax + t \quad (2.2)$$

La matriz A puede descomponerse en el producto de transformaciones más simples[16],

$$A = RSC$$

R es una matriz de rotación, S es una matriz de escalamiento y C es una matriz de cizallamiento. La factorización de la matriz A no se realiza necesariamente en ese orden.

En esta caso, la transformación queda parametrizada por $D(D+1)$ valores, siendo D la dimensión del vector x .

Las transformaciones afines son de gran utilidad en registro de imágenes debido a la composición de transformaciones que engloba.

2.2.2 Transformaciones no Paramétricas

De manera general, las transformaciones rígidas y afines son insuficientes para realizar el registro entre dos imágenes, sobre todo cuando se trata de imágenes que no pertenecen a una misma escena y que solamente comparten algunas estructuras o rasgos similares. En tales casos es preciso utilizar un modelo más general que permita explicar de manera adecuada la transformación de la imagen de referencia en la imagen objetivo. Dicho modelo es el siguiente

$$T(\mathbf{x}_i) = \mathbf{x}_i + \mathbf{t}_i, \text{ para } i = 1, \dots, N$$

Este modelo aplica un vector de traslación \mathbf{t}_i distinto a cada punto \mathbf{x}_i , por lo que si N es el número de puntos de la imagen y D es la dimensión, el registro de ambas imágenes contendrá ND incógnitas que deben ser estimadas. El campo de desplazamientos \mathbf{t} , puede ser modelado por medio de polinomios o por splines [17], reduciendo de esta manera la magnitud del problema al limitar el registro a la búsqueda de los coeficientes de los polinomios o los coeficientes de las funciones base de splines.

2.3 Modalidades del Registro de Imágenes

El registro de imágenes se basa fuertemente en explotar las características de la función de transferencia de tonos, expresando generalmente una función sobre el espacio de los parámetros de la transformación que debe ser optimizada.

En la presente sección se analiza la clasificación del registro de acuerdo a las características de la función de transferencia de tonos.

2.3.1 Registro Monomodal

Cuando la función de transferencia de tonos $\phi(\mathbf{I})$ es igual o muy similar a la identidad, hablamos de registro monomodal. Esto significa que el registro se realiza entre imágenes obtenidas por la misma técnica, generalmente de la misma escena en intervalos de tiempo

diferentes, por lo que podemos esperar que los componentes de la escena hayan cambiado su ubicación.

2.3.2 Registro Multimodal

Si la función de transferencia de tonos es distinta a la identidad, hablamos de registro multimodal. Lo anterior significa que el registro se realiza entre imágenes obtenidas mediante técnicas distintas. La aplicación del registro multimodal en el campo de la medicina es extenso y su importancia radica en que la información obtenida a través de técnicas distintas es complementaria, en [12] puede encontrarse una descripción detallada de tales aplicaciones.

2.4 Esquema Multiescala

El desempeño de los algoritmos de registro puede ser mejorado notablemente utilizando un esquema multiescala. Dicho esquema produce un arreglo piramidal de imágenes aplicando operaciones de filtrado y submuestreo a las imágenes originales. El filtrado puede realizarse mediante la convolución de la imagen con un kernel gaussiano, por ejemplo. En cada nivel del arreglo piramidal identificamos un par de imágenes producto de las imágenes objetivo y de referencia cada una, en las cuales aplicamos el algoritmo de registro, comenzando en el nivel más alto y transportando la solución de cada nivel al nivel inmediato inferior, terminando en el más bajo, en el cual las imágenes originales son registradas, fig. 2-1. Deberá tenerse cuidado de reescalar los parámetros que son afectados por la escala al transportar las soluciones entre niveles, como puede ser el caso de los parámetros relacionados con la traslación.

El esquema multiescala permite tratar con problemas de ruido y de grandes transformaciones, ya que produce la eliminación de estructuras débiles y además es posible trabajar con un número reducido de puntos en los niveles más altos [13].

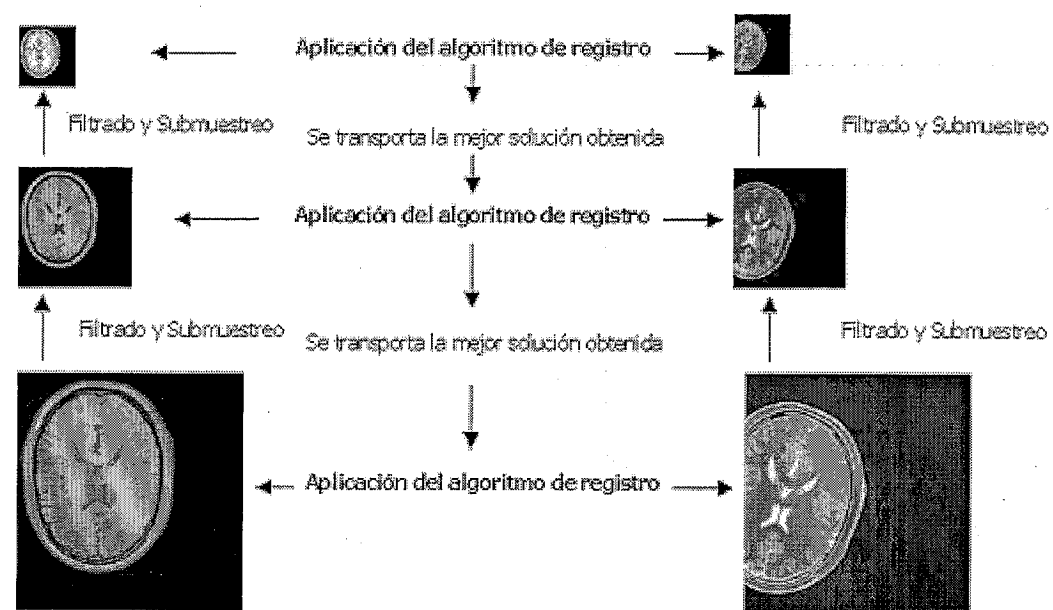


Figura 2-1: En esta figura se muestra el funcionamiento general de un esquema multiescala en el registro de imágenes.

Capítulo 3

Estrategias Evolutivas

3.1 Antecedentes

Las *estrategias evolutivas* son técnicas de optimización que imitan los procesos evolutivos de las especies en la naturaleza. Están basadas en una población de individuos (cada uno de los cuales representa una posible solución al problema de optimización) sobre la cual se aplican operadores de recombinación, mutación y selección, para dirigir la búsqueda hacia el óptimo.

Los inicios de esta técnica se remontan a la década de los 60's y se debió a los trabajos de Rechenberg [18] sobre hidrodinámica en la Universidad Técnica de Berlín. Rechenberg aplicó un esquema de *mutación-selección* sobre un único individuo, al problema de diseño de la forma de una tubería, buscando minimizar la pérdida de energía del flujo a través de ella. En esta primera versión de las estrategias evolutivas, un individuo representaba un conjunto de posiciones que determinaban la geometría del conducto. Bajo la idea de que las mutaciones pequeñas ocurren con una mayor probabilidad, se inducían sobre el individuo cambios mediante la adición de un número aleatorio discreto de una distribución binomial con media cero y desviación estándar fija (a la cual denominó *tamaño de paso*) a través de todo el proceso, obteniendo un nuevo individuo, el cual sustituía al anterior en caso de que representara una mejor solución.

Posteriormente Rechenberg aplicó su estrategia a problemas de optimización continua realizando la mutación mediante la adición de un número aleatorio generado a través de una distribución normal con media cero y desviación estándar fija.

Mediante el estudio de la velocidad de convergencia en función del valor del tamaño de paso sobre varias funciones de prueba, Rechenberg encontró que la máxima velocidad de convergencia se daba con desviaciones estándar que producían mutaciones exitosas con un 20% de probabilidad, entendiendo una mutación exitosa como aquella que superaba la solución del individuo del cual era generada, con lo que aplicó actualizaciones al tamaño de paso si después de un cierto número de generaciones el porcentaje de mutaciones exitosas era distinto al 20%, incrementando el tamaño de paso en un factor fijo en caso de que el porcentaje fuera mayor, o decrementándolo por el mismo factor en caso de que el porcentaje fuera menor. Así mismo, Rechenberg propuso el concepto de población, al utilizar un conjunto de μ padres, $\mu > 1$, para la generación de un hijo, introduciendo a la vez el operador de recombinación en estrategias evolutivas.

El impulso definitivo de las estrategias evolutivas se produjo a través de Hans Paul Schwefel, quien visualizó el tamaño de paso como parte de la información de cada individuo, de manera que pudiera ser actualizado por medio de los operadores de recombinación y mutación, logrando que el mismo proceso evolutivo determinara el valor del tamaño de paso adecuado. Schwefel además introdujo la idea de generar una población de λ hijos, $\lambda > 1$, a partir de la población de padres, derivando en la versión más actual de las estrategias evolutivas, la cual se describe en la siguiente sección.

3.2 Descripción de las Estrategias Evolutivas.

De acuerdo con [19], las estrategias evolutivas se definen como la tupla

$$EE = \{I, \Phi, \mu, \lambda, r, m, s, t\}$$

en donde

I = Espacio de individuos.

Φ = Función de aptitud.

μ = Tamaño de la población de padres.

λ = Tamaño de la población de hijos.

r = Operador de recombinación.

m = Operador de mutación.

s = Operador de selección.

t = Criterio de paro del algoritmo.

Las estrategias evolutivas son aplicadas al problema de optimización de una función objetivo f , buscando el vector de parámetros \mathbf{x}^* tal que

$$\mathbf{x}^* = \arg \max f(\mathbf{x}), \quad f : R^n \rightarrow R$$

3.2.1 Codificación

Cada individuo es codificado mediante un conjunto de 2 vectores reales n -dimensionales siendo n el número de variables a optimizar, de la siguiente manera

$$I = \mathbf{a} = \{\mathbf{x}, \boldsymbol{\sigma}\} = \{x_1, \dots, x_n, \sigma_1, \dots, \sigma_n\}$$

\mathbf{x} = Vector de *variables objetivo*. Contiene cada una de las variables a optimizar.

$\boldsymbol{\sigma}$ = Vector de *variables de control*. Generalmente se asigna una variable de control a cada variable objetivo. Se utilizan para controlar el tamaño del cambio al aplicar el operador de mutación.

A partir de un conjunto de individuos, definimos las poblaciones de *padres* e *hijos* como

$$\text{Población de Padres} = P_p = \{\mathbf{a}_1, \dots, \mathbf{a}_\mu\}$$

$$\text{Población de Hijos} = P_h = \{\mathbf{a}'_1, \dots, \mathbf{a}'_\lambda\}$$

3.2.2 Función de Aptitud

La *función de aptitud* Φ asigna un número real, llamado *aptitud*, a cada individuo, de acuerdo a la calidad de la solución que éste representa en el problema de optimización, de manera que los mejores individuos tengan un valor de aptitud más alto que los peores individuos.

Generalmente la función de aptitud es igual a la función objetivo, cumpliéndose la siguiente igualdad

$$\Phi(\mathbf{a}) = f(\mathbf{x})$$

siendo \mathbf{x} el vector de variables objetivo del individuo \mathbf{a} .

3.2.3 Recombinación

La recombinación puede ser descrita como la aplicación repetida (λ veces) de la composición de los operadores *el* y *cr*, *elección* y *cruza* respectivamente, sobre la población de padres. En cada aplicación se genera un nuevo hijo.

El operador de elección, $el : \mathbf{I}^\mu \rightarrow \mathbf{I}^\rho$, elige de manera aleatoria ρ individuos mediante una distribución uniforme sobre la población de padres, de modo que cada individuo de tal población tiene la misma posibilidad de ser elegido, estos individuos serán la base para generar un nuevo hijo

$$el(\mathbf{P}_p) = \{\mathbf{a}_1, \dots, \mathbf{a}_\rho\}$$

El valor de ρ determina dos modalidades distintas para la recombinación, ya que si $\rho = 2$, la recombinación se denomina *sexual*, y si $\rho > 2$, la recombinación se denomina *panmítica*.

El operador de cruza permite mezclar la información contenida en la codificación del conjunto de individuos elegidos por la aplicación de $el(\mathbf{P}_p)$ para producir un nuevo hijo.

De manera que

$$cr(el(\mathbf{P}_p)) = \mathbf{a}' = \{\mathbf{x}', \boldsymbol{\sigma}'\} = \{x'_1, \dots, x'_n, \sigma'_1, \dots, \sigma'_n\}$$

Si denominamos c'_i a la componente i -ésima del vector de variables objetivo o de variables de control del nuevo hijo, $c'_i = x'_i$ ó σ'_i , la cruza se realiza de la siguiente manera para el caso de la recombinación sexual

$$c'_i = \begin{cases} c_{i,1} \text{ si } \chi_i \geq 0.5 \text{ ó } c_{i,2} \text{ si } \chi_i < 0.5 & \text{recombinación sexual discreta} \\ \frac{1}{2}(c_{i,1} + c_{i,2}) & \text{recombinación sexual intermedia} \\ \chi_i c_{i,1} + (1 - \chi_i) c_{i,2} & \text{recombinación sexual general intermedia} \end{cases}$$

$$i = 1, \dots, n$$

$c_{i,1}$ = Componente i -ésima del primer padre.

$c_{i,2}$ = Componente i -ésima del segundo padre.

χ_i = Número aleatorio generado de una distribución uniforme $U[0, 1]$

Lo anterior aplica también para la recombinación panmítica con la diferencia de que se utiliza un padre fijo a través de todo el proceso y se escoge un nuevo padre para generar cada componente del individuo mutado, por lo que generalmente $\rho = 2n + 1$.

La recombinación permite explorar ampliamente el espacio de búsqueda al generar hijos distribuidos entre la población de padres, como puede observarse en la figura 3-1, en la cual se ha marcado con un punto a cada individuo de la población de padres y mediante cruces a los hijos generados por la recombinación.

3.2.4 Mutación

El operador de mutación actúa sobre la población de hijos, $m : \mathbf{I}^\lambda \rightarrow \mathbf{I}^\lambda$, aplicando repetidamente su versión reducida $m' : \mathbf{I} \rightarrow \mathbf{I}$, sobre cada individuo, induciendo cambios aleatorios en cada componente de las variables objetivo y de control.

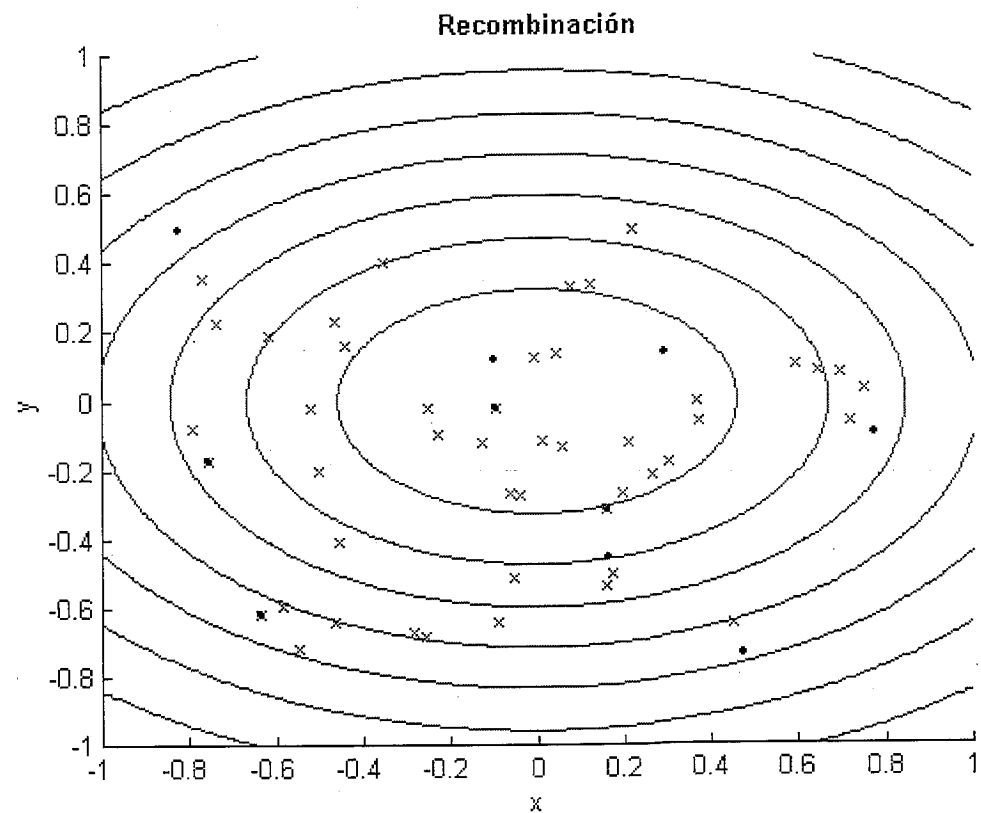


Figura 3-1: En la figura se representan con puntos los individuos de la población de padres y con cruces los hijos.

Inicialmente m' se aplica a las variable de control y los valores obtenidos se utilizan para mutar las variables objetivo asociadas, generando el individuo $a' = \{x', \sigma'\} = \{x'_1, \dots, x'_n, \sigma'_1, \dots, \sigma'_n\}$, a partir del individuo $a = \{x, \sigma\} = \{x_1, \dots, x_n, \sigma_1, \dots, \sigma_n\}$, mediante el siguiente proceso

$$\sigma'_i = \sigma_i \exp[\tau N(0, 1) + \tau N_i(0, 1)]$$

$$x'_i = x_i + N(0, \sigma_i)$$

$$i = 1, \dots, n$$

$$\tau = \frac{1}{\sqrt{2\sqrt{n}}}$$

$$\tau = \frac{1}{\sqrt{2n}}$$

$N(0, 1)$ es un número aleatorio de una distribución normal con media cero, que se genera al inicio de la mutación del individuo y que permanece fijo al mutar cada componente.

$N_i(0, 1)$ es un número aleatorio de una distribución normal con media cero, generado para cada componente.

La utilización de números aleatorios de una distribución normal con media cero como argumento para la mutación de los individuos, asegura que los cambios pequeños sean producidos con mayor probabilidad que los cambios grandes. De la misma forma, la mutación de las variables de control por un proceso multiplicativo mediante una exponencial, asegura que el valor de dichas variables sea siempre positivo si se inicia con valores positivos, no obstante, estos valores pueden llegar a ser arbitrariamente pequeños. Otro punto importante es que el valor de las variables, tanto de control como objetivo, puede ser aumentado o reducido por el mismo factor con la misma probabilidad, asegurando la neutralidad del proceso de mutación.

En la figura 3-2, puede observarse la acción del operador de mutación sobre una población arbitraria.

3.2.5 Selección

El proceso de selección permite generar la nueva población de padres, escogiendo para ello a los μ mejores individuos de acuerdo a su valor de aptitud. En este sentido, encontramos dos versiones del operador de selección, cada una de las cuales origina una estrategia evolutiva distinta.

La primer versión selecciona a los μ mejores individuos de la población generada por la unión de padres e hijos, $s_{(\mu+\lambda)} : \mathbf{I}^{\mu+\lambda} \rightarrow \mathbf{I}^{\mu}$, su aplicación da lugar a la denominada *Estrategia Evolutiva* $(\mu + \lambda)$. La segunda versión opera sobre la población de hijos sola-

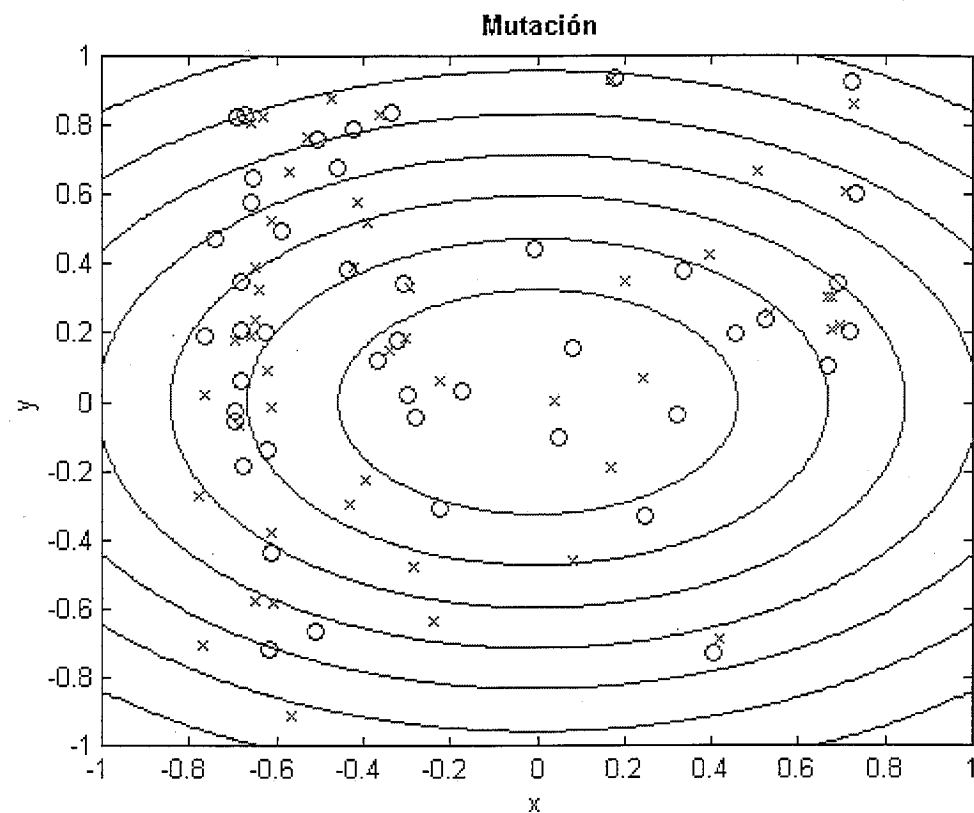


Figura 3-2: Representación de la población de hijos antes (cruces) y después (círculos) de la mutación.

mente, $s_{(\mu,\lambda)} : \mathbf{I}^\lambda \rightarrow \mathbf{I}^\mu$, escogiendo a los μ mejores individuos entre ellos, dando lugar a la denominada *Estrategia Evolutiva* (μ, λ) , en la que necesariamente $\mu \leq \lambda$.

En la *Estrategia Evolutiva* $(\mu + \lambda)$, un individuo de la población de padres sobrevive indefinidamente, hasta que se genera un número suficiente de hijos que lo superen, o por lo menos, lo igualen en aptitud, mientras que en la *Estrategia Evolutiva* (μ, λ) un padre vive solamente una generación.

Aunque a primera vista pareciera que la *Estrategia Evolutiva* $(\mu + \lambda)$ ofrece más ventajas que la versión (μ, λ) , las ventajas de esta última aparecen cuando el óptimo cambia a través del tiempo o cuando la función de aptitud es evaluada con la presencia

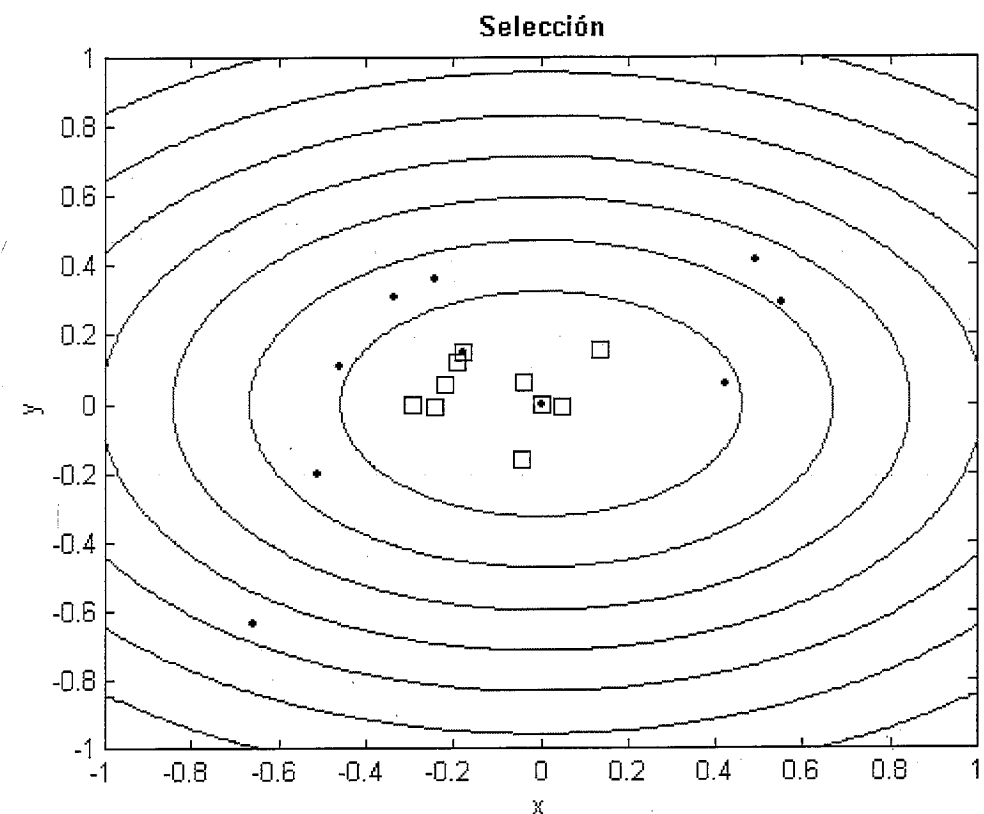


Figura 3-3: Poblaciones de padres antes (puntos) y después (cuadros) de aplicar la selección.

de ruido numérico, ya que asegura la no supervivencia de individuos que obtengan un valor de aptitud alto en condiciones de evaluación especiales.

En la figura 3-3, se pueden observar las poblaciones de padres antes y después de aplicar el operador de selección. El óptimo se encuentra en el centro de la imagen y los nuevos padres se han distribuido más cerca de él en relación a los anteriores.

3.2.6 Algoritmo General

El algoritmo general de una estrategia evolutiva comienza con la creación aleatoria de una población de padres. De manera iterativa, la población de padres genera hijos a

partir del operador de recombinación, los hijos son mutados y a través de la selección se generan nuevos padres.

- 1.- Generar $\mathbf{P}_{p,0} = \{\mathbf{a}_1, \dots, \mathbf{a}_\mu\}$
- 2.- Evaluar Aptitud $\Phi(\mathbf{P}_{p,0}) = \{\Phi(\mathbf{a}_1), \dots, \Phi(\mathbf{a}_\mu)\}$
- 3.- Iniciar contador de generaciones, $g = 0$.
- 4.- Mientras ($t(\mathbf{P}_{p,g}) = falso$)
 - Recombinar $r(\mathbf{P}_{p,g}) \rightarrow \mathbf{P}_{h,g} = \{\mathbf{a}_1, \dots, \mathbf{a}_\lambda\}$
 - Mutar $m(\mathbf{P}_{h,g}) \rightarrow \mathbf{P}'_{h,g} = \{\mathbf{a}'_1, \dots, \mathbf{a}'_\lambda\}$
 - Seleccionar
 - $s_{(\mu+\lambda)}(\mathbf{P}_{p,g} \cup \mathbf{P}'_{h,g}) = \mathbf{P}_{p,g+1}$
 - ó
 - $s_{(\mu,\lambda)}(\mathbf{P}'_{h,g}) = \mathbf{P}_{p,g+1}$
 - Incrementar generaciones, $g = g + 1$.
- 5.- Fin

El resultado del algoritmo es el individuo con mayor valor de aptitud generado durante todo el proceso evolutivo.

La población inicial puede ser creada alrededor de un punto de arranque, aplicando mutaciones a un individuo cuyas variables objetivo sean los valores de dicho punto, o puede distribuirse aleatoriamente en un espacio determinado. Cabe señalar que a diferencia de los algoritmos genéticos, no es necesario delimitar con exactitud el espacio de búsqueda, lo cual constituye una ventaja en los casos en que no se conoce con certeza la región en que se encuentra el óptimo.

Es recomendable utilizar una relación entre los tamaños de las poblaciones de padres e hijos, $\frac{\mu}{\lambda}$, entre $\frac{1}{5}$ y $\frac{1}{7}$, ya que esos valores aseguran una presión de selección suficientemente grande, lo cual repercute en el éxito de su aplicación. La elección del valor de μ , dependerá de qué tan ampliamente se desea explorar el espacio de parámetros a optimizar.

Para las variables de control, Schwefel recomienda sean iniciadas mediante la relación

$$\sigma_{i,0} = \frac{\Delta x_i}{\sqrt{n}}$$

$$i = 1, \dots, n$$

en la que Δx_i representa la distancia estimada entre el punto de arranque y el óptimo para la variable x_i .

El criterio de paro del algoritmo puede determinarse a partir de un número fijo de generaciones, un valor de aptitud alcanzado por algún individuo o una diferencia pequeña entre el individuo con mayor aptitud y el individuo con menor aptitud.

Capítulo 4

Registro de Imágenes Basado en Intensidad

Cuando la función de transferencia de tonos entre las imágenes a registrar es la identidad, $\Phi(I(\mathbf{x})) = I(\mathbf{x})$, como sucede en el caso de registro monomodal, la ecuación 2.1 se reduce a

$$I_1[T(\mathbf{x})] = I_2(\mathbf{x})$$

y el registro puede realizarse basándose en la comparación de las intensidades entre cada punto de las imágenes objetivo y de prueba. Dicha comparación, generalmente es hecha a través de una diferencia, asociando un valor *residual* sobre cada punto de las imágenes

$$r_i = I_1[T(\mathbf{x}_i)] - I_2(\mathbf{x}_i)$$

$$i = 1, \dots, N.$$

En general, las estrategias de registro buscarán encontrar la transformación $T(\mathbf{x})$, mediante la optimización de una función objetivo, que minimice el residuo de cada uno de los puntos.

4.1 Suma de Diferencias al Cuadrado

La *Suma de Diferencias al Cuadrado*, SSD por sus siglas en inglés, permite formular una función objetivo que depende de la suma del cuadrado de los residuos de cada punto

$$f(T) = \frac{1}{2} \sum_{i=1}^N r_i^2 = \frac{1}{2} \sum_{i=1}^N [I_2[T(\mathbf{x}_i)] - I_2(\mathbf{x}_i)]^2$$

Si definimos el vector de residuos como $\mathbf{r} = [r_1 \ r_2 \ \dots \ r_N]$, la ecuación anterior puede reescribirse

$$f(T) = \frac{1}{2} \|\mathbf{r}\|^2$$

$\|\cdot\|$ denota la norma euclídeana.

En el caso de que la transformación $T(\mathbf{x})$ sea paramétrica, como supondremos en el resto de este trabajo, la función objetivo dependerá directamente de tales parámetros

$$f(\boldsymbol{\theta}) = \frac{1}{2} \sum_{i=1}^N r(\boldsymbol{\theta})_i^2 = \frac{1}{2} \sum_{i=1}^N [I_2[T(\boldsymbol{\theta}, \mathbf{x}_i)] - I_2(\mathbf{x}_i)]^2 \quad (4.1)$$

$\boldsymbol{\theta} = \{\theta_1, \dots, \theta_m\}$, es el vector de parámetros que define la transformación, $\boldsymbol{\theta} \in R^m$.

Esta formulación es equivalente al empleo de un esquema de mínimos cuadrados, existiendo una serie de métodos tradicionales que se aplican con gran éxito en la optimización de este tipo de funciones, una descripción detallada puede encontrarse en [20].

4.1.1 Método de Gauss-Newton

El método de *Gauss-Newton* es una técnica de optimización iterativa, basada en la generación de una sucesión de parámetros $\{\boldsymbol{\theta}^0, \boldsymbol{\theta}^1, \dots\}$, a partir de un punto inicial $\boldsymbol{\theta}^0$, mediante la relación

$$\boldsymbol{\theta}^{k+1} = \boldsymbol{\theta}^k + \alpha^k \mathbf{p}^k \quad (4.2)$$

$\mathbf{p}^k \in R^m$, dirección de actualización.

$\alpha^k \in R$, tamaño de paso.

La elección de la dirección de actualización \mathbf{p}^k , se realiza a través de un modelo cuadrático que se obtiene mediante la expansión en serie de Taylor de la función objetivo

$$f(\boldsymbol{\theta}^k + \mathbf{p}^k) \approx f(\boldsymbol{\theta}^k) + (\mathbf{p}^k)^T \nabla f(\boldsymbol{\theta}^k) + \frac{1}{2} (\mathbf{p}^k)^T \nabla^2 f(\boldsymbol{\theta}^k) \mathbf{p}^k$$

$\nabla f(\boldsymbol{\theta}^k) = \sum_{j=1}^m r_j \nabla r_j$, vector gradiente de la función objetivo.

$\nabla^2 f(\boldsymbol{\theta}^k) = \sum_{j=1}^m \nabla r_j \nabla r_j^T + \sum_{j=1}^m r_j \nabla^2 r_j^T$, matriz Hessiana de la función objetivo.

$\nabla r_i = \left[\frac{\delta r_i}{\delta \theta_1} \ \frac{\delta r_i}{\delta \theta_2} \ \dots \ \frac{\delta r_i}{\delta \theta_m} \right]^T$, vector de derivadas del residuo i -ésimo.

Utilizando como aproximación al Hessiano

$$\nabla^2 f(\boldsymbol{\theta}^k) \approx \sum_{j=1}^m \nabla r_j \nabla r_j^T = \mathbf{J}^T \mathbf{J}$$

\mathbf{J} , matriz Jacobiana.

En la relación anterior se han eliminado los términos de segundas derivadas de los residuos en la matriz Hessiana.

El modelo resultante es el siguiente

$$m(\mathbf{p}^k) = f(\boldsymbol{\theta}^k) + (\mathbf{p}^k)^T \nabla f(\boldsymbol{\theta}^k) + \frac{1}{2} (\mathbf{p}^k)^T (\mathbf{J}^T \mathbf{J}) \mathbf{p}^k$$

Calculando el gradiente del modelo e igualándolo a cero, encontramos el vector de dirección que lo minimiza

$$\mathbf{p}^k = -(\mathbf{J}^T \mathbf{J})^{-1} \nabla f(\boldsymbol{\theta}^k)$$

El tamaño de paso debe escogerse de manera que asegure el máximo descenso en la función objetivo, y puede mostrarse [20] que en los casos en que los términos de segundas derivadas del Hessiano que han sido eliminados son despreciables, un valor de α^k igual a

la unidad es adecuado, por lo que finalmente el esquema iterativo 4.2 se reescribe

$$\theta^{k+1} = \theta^k - (\mathbf{J}^T \mathbf{J})^{-1} \nabla f(\theta^k)$$

La aplicación del método de Gauss-Newton al problema de registro paramétrico, se realiza a través de la optimización de la suma de diferencias al cuadrado, en la que identificamos el vector gradiente de la función 4.1

$$\nabla f(\theta^k) = \sum_{i=1}^N r_i \nabla r_i$$

y la matriz Jacobiana

$$\mathbf{J}^T = \begin{bmatrix} \nabla r_1 & \nabla r_2 & \dots & \nabla r_N \end{bmatrix}$$

El algoritmo de registro se describe a continuación

- 1.- Iniciar el vector de parámetros θ^0 .
- 2.- Iniciar iteraciones en cero $t = 0$.
- 3.- Mientras (*criterio de paro = falso*)

Construir $\mathbf{I}_1[T(\theta^t, \mathbf{x})]$

Calcular vector de residuos $\mathbf{r}^t = \{r_1^t, \dots, r_N^t\}$

Calcular la matriz $\mathbf{J}^T \mathbf{J} = \begin{bmatrix} \nabla r_1^t & \nabla r_2^t & \dots & \nabla r_N^t \\ \nabla r_2^t \\ \vdots \\ \nabla r_N^t \end{bmatrix}$

Calcular $\nabla f(\theta^t) = \mathbf{J}^T \mathbf{r}^t$

Actualizar parámetros $\theta^{t+1} = \theta^t - (\mathbf{J}^T \mathbf{J})^{-1} \mathbf{J}^T \mathbf{r}^t$

Incrementar contador de iteraciones $t = t + 1$

- 4.- Fin.

El criterio de paro puede establecerse en relación a la magnitud del vector residual, a la magnitud de los cambios producidos en el vector de parámetros o mediante un número fijo de iteraciones.

Generalmente el vector de parámetros inicial es el que determina la transformación identidad, de manera que la primer imagen de prueba consistirá en la imagen de referencia

$$\mathbf{I}_1[T(\theta^0, \mathbf{x})] = \mathbf{I}_1(\mathbf{x})$$

De acuerdo a lo comentado en la sección 2.4 puede utilizarse un esquema multiescala para el registro mediante Gauss-Newton, con lo que se aplica el algoritmo en el nivel más alto de la pirámide gaussiana, iniciando con la transformación identidad y el resultado de cada nivel se utiliza como punto inicial del algoritmo en el nivel inmediato inferior, hasta aplicarlo en las imágenes originales.

Construcción de la Imagen de Prueba

El cálculo del vector residual, requiere la construcción de la imagen $\mathbf{I}_1[T(\mathbf{x})]$ en cada iteración. La imagen $\mathbf{I}_1(\mathbf{x})$ se encuentra definida solamente para un conjunto discreto de puntos que corresponden a valores enteros del vector de coordenadas \mathbf{x} , por lo que la evaluación de la intensidad en el punto $\mathbf{x}' = T(\mathbf{x})$, debe realizarse por medio de interpolación numérica sobre las intensidades de los puntos enteros vecinos a él, ya que generalmente será un vector real D - dimensional.

En este trabajo se utiliza interpolación por medio de splines [14], [15]. La evaluación de la intensidad de la imagen de referencia en el punto $\mathbf{x}' = \{x_1', \dots, x_D\}$ se realiza mediante la siguiente relación

$$\mathbf{I}_1(\mathbf{x}') = \sum_{k=1}^N c(\mathbf{x}_k) \beta_D^n(\mathbf{x}' - \mathbf{x}_k)$$

en la que

$c(\mathbf{x}_k)$ es un coeficiente real que se asocia al punto k -ésimo.

$\beta_D^n(\mathbf{x}' - \mathbf{x}_k)$ es una función base de spline D - dimensional y de grado n , centrada en el punto k -ésimo, la cual puede obtenerse a partir de la multiplicación de funciones base de splines 1 - dimensionales de la siguiente manera

$$\beta_D^n(\mathbf{x}' - \mathbf{x}_k) = \prod_{i=1}^D \beta_1^n(x_i' - x_{k,i})$$

Generalmente $n = 2$ ó $n = 3$, con lo que las funciones base de splines quedan definidas de la siguiente manera

$$\beta_1^2(x) = \begin{cases} \frac{1}{2}(-2x^2 + \frac{3}{2}), & |x| < \frac{1}{2} \\ \frac{1}{2}(x^2 - 3|x| + \frac{9}{4}), & \frac{1}{2} \leq |x| < \frac{3}{2} \\ 0, & |x| \geq \frac{3}{2} \end{cases}$$

$$\beta_1^3(x) = \begin{cases} \frac{2}{3} - x^2 + \frac{|x|^3}{2}, & |x| < 1 \\ \frac{1}{6}(2 - |x|)^3, & 1 \leq |x| < 2 \\ 0, & |x| \geq 2 \end{cases}$$

El cálculo de los coeficientes se realiza mediante un proceso que recorre ordenadamente cada una de las dimensiones de la imagen. Para la dimensión i -ésima, se calculan los coeficientes recorriendo dos veces todos los puntos de la imagen. En el primer recorrido se calculan los valores

$$c_i^+(\mathbf{x}_k) = s_i(\mathbf{x}_k) + z_1 c_i^+(\mathbf{x}_{k-1}), \quad k = 2, \dots, N$$

y en el segundo

$$c_i^-(\mathbf{x}_k) = z_1 [c_i^-(\mathbf{x}_{k+1}) - c_i^+(\mathbf{x}_k)], \quad k = N - 1, \dots, 1$$

El primer punto de estas relaciones recursivas se estima de la siguiente manera

$$c_i^+(\mathbf{x}_1) = \sum_{k=1}^{K_0} s_i(\mathbf{x}_k) z_1^k$$

$$c_i^-(\mathbf{x}_N) = \frac{z_1}{1 - z_1^2} [c_i^+(\mathbf{x}_N) + z_1 c_i^+(\mathbf{x}_{N-1})]$$

$K_0 > \log(\epsilon) / \log |z_1|$, ϵ es el nivel deseado de precisión.

y finalmente

$$c_i(\mathbf{x}_k) = 6c_i^-(\mathbf{x}_k)$$

En las relaciones anteriores

$$s_i(\mathbf{x}_k) = \begin{cases} I_1(\mathbf{x}_k), & \text{si } i = 1. \\ c_{i-1}(\mathbf{x}_k), & \text{si } i \neq 1. \end{cases}$$

$$z_1 = \begin{cases} -3 + 2\sqrt{2}, & \text{si } n = 2. \\ -2 + \sqrt{3}, & \text{si } n = 3. \end{cases}$$

4.2 Estrategias Evolutivas Aplicadas a Registro Basado en Intensidad

En esta sección se presenta una propuesta de un algoritmo de registro, el cual utiliza las estrategias evolutivas como técnica de optimización sobre los residuos en los puntos de las imágenes a registrar.

Para determinar la aplicación de las estrategias evolutivas a cualquier problema es necesario identificar cada uno de los componentes de la sección 3.2, realizado esto, la utilización del algoritmo descrito en la sección 3.2.6 queda definida.

4.2.1 Codificación

Se utilizan como variables objetivo los mismos parámetros de la transformación, de manera que cada individuo representa una transformación distinta. Se asigna una variable de control a cada parámetro, por lo que cada individuo será representado por un vector *2m-dimensional*

$$\mathbf{I} = \mathbf{a} = \{\theta_1, \dots, \theta_m, \sigma_1, \dots, \sigma_m\}$$

4.2.2 Función de Aptitud

Tomando en cuenta que las estrategias evolutivas trabajan bajo un esquema de maximización, la función de aptitud se eligió de manera que su valor fuera inversamente proporcional al residual

$$\Phi(\mathbf{a}) = \frac{1}{1 + \frac{1}{N} \sum_{\mathbf{x}_i \in CI} |r(\mathbf{x}_i)|} = \frac{1}{1 + \frac{1}{N} \sum_{\mathbf{x}_i \in CI} |I_1[T(\theta, \mathbf{x}_i)] - I_2(\mathbf{x}_i)|}$$

θ , vector de variables objetivo del individuo \mathbf{a} .

CI , conjunto de puntos utilizado para realizar la comparación entre las imágenes.

N , cardinalidad del conjunto CI .

En lugar de utilizar todos los puntos de la imagen para realizar la comparación, lo que haría la evaluación de la función de aptitud muy cara, se utiliza un conjunto reducido de puntos de interés CI , que pueden escogerse equiespaciados sobre la imagen de referencia o aleatoriamente distribuidos sobre toda la imagen mediante una distribución uniforme en las coordenadas. En este último caso, el conjunto puede renovarse al transcurrir un número fijo de generaciones, dando buenos resultados al trabajar con imágenes ruidosas.

El número de puntos utilizados para evaluar la aptitud, dependerá de la variabilidad de las intensidades de las imágenes a registrar y del tamaño de las mismas. En la práctica hemos obtenido buenos resultados utilizando alrededor del 0.5% de los puntos de la imagen.

La función de aptitud asigna a cada individuo un valor en el intervalo $(0, 1]$, variando de la *no correspondencia* hasta la *correspondencia perfecta*, evaluada sobre el conjunto CI .

Es posible relacionar un error promedio en el registro, en base a la función de aptitud

$$error\ promedio(\mathbf{a}) = \frac{1}{\Phi(\mathbf{a})} - 1$$

lo cual puede utilizarse como un criterio de paro del algoritmo, al detener la búsqueda cuando algún individuo alcance un valor de aptitud que tenga asociado un error promedio menor a una tolerancia definida por el usuario. En la valoración de este error promedio, se debe tomar en cuenta el rango de las intensidades de las imágenes a registrar.

4.2.3 Recombinación, Mutación y Selección

En este trabajo, se utilizó recombinación sexual intermedia generalizada, tanto para las variables objetivo como para las variables de control. La mutación se realiza bajo el esquema de la sección 3.2.4. Se aplicó selección $(\mu + \lambda)$, ya que en la práctica se obtuvieron mejores resultados que con la estrategia (μ, λ) .

4.2.4 Esquema Multiescala

La aplicación de las estrategias evolutivas al problema de registro puede realizarse bajo un esquema multiescala, ejecutando el algoritmo en cada nivel de la pirámide gaussiana. En el nivel más alto, se siembra en la población un individuo que represente la transformación identidad o el mejor punto de partida de que se disponga, y el resto de los individuos se genera sumando al vector de variables objetivo del individuo sembrado, un número aleatorio de una distribución uniforme que abarque el intervalo de búsqueda

$$\theta = \{\theta_{s,1} + U[-a_1, a_1], \dots, \theta_{s,m} + U[-a_m, a_m]\} \quad (4.3)$$

$\theta_{s,i}$, parámetro i -ésimo del individuo sembrado.

$U[-a_i, a_i]$, número aleatorio de una distribución uniforme con ancho a_i centrada en cero.

El valor de a_i se escoge de acuerdo a las dimensiones del espacio de búsqueda que se quiera cubrir.

En cada nivel la población es renovada, sembrando el individuo con mayor aptitud que haya sido generado en el nivel anterior y creando el resto de la población uniformemente distribuida en un espacio centrado en el individuo sembrado, mediante la relación 4.3. En cada nivel el valor de a_i puede reducirse con respecto a su valor en el nivel anterior.

El algoritmo puede detenerse al generar un individuo con suficiente valor de aptitud, cuando la diferencia de aptitud entre el mejor y el peor individuo sea pequeña o mediante un número fijo de generaciones.

Utilizar un esquema multiescala para la aplicación de las estrategias evolutivas es de gran importancia pues las imágenes en los niveles más altos de la pirámide gaussiana están compuestas por un número reducido de puntos, y la evaluación de la función de aptitud puede hacerse con un porcentaje muy alto de puntos de la imagen, lo cual hace la evaluación más significativa. Esto permite orientar fuertemente la búsqueda en los niveles altos y refinarla en los niveles bajos.

4.3 Comparación entre Algoritmos de Gauss-Newton y Estrategias Evolutivas Aplicados a Registro Basado en Intensidad

La comparación entre los algoritmos descritos anteriormente, se realizó a través de transformaciones afines sintéticas aplicadas a diferentes imágenes bidimensionales. Para este

caso la transformación de coordenadas puede escribirse

$$T(x, y) = \begin{bmatrix} \theta_1 & \theta_2 \\ \theta_3 & \theta_4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} \theta_5 \\ \theta_6 \end{bmatrix}$$

y como se señaló en la sección 2.2.1, esta transformación es la composición de una rotación, un escalamiento, un cizallamiento y una traslación

$$T(x, y) = \mathbf{RECx} + \mathbf{t} \quad (4.4)$$

$$\mathbf{R} = \begin{bmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{bmatrix}, \text{ matriz de rotación de un ángulo } \alpha.$$

$$\mathbf{E} = \begin{bmatrix} e_x & 0 \\ 0 & e_y \end{bmatrix}, \text{ matriz de escalamiento con factores } e_x, e_y.$$

$$\mathbf{C} = \begin{bmatrix} 1 & c_x \\ 0 & 1 \end{bmatrix} \text{ ó } \begin{bmatrix} 1 & 0 \\ c_y & 1 \end{bmatrix}, \text{ matriz de cizallamiento sobre eje } x \text{ o sobre eje } y \text{ resp..}$$

$$\mathbf{t} = \begin{bmatrix} t_1 \\ t_2 \end{bmatrix}, \text{ vector de traslación.}$$

Las pruebas se realizaron generando valores aleatorios para ángulos, factores de escala, cizallamiento y traslación mediante una distribución uniforme sobre intervalos numéricos cuya amplitud se fue aumentando gradualmente, y construyendo las transformaciones afines determinadas por los valores generados. Para cada intervalo, se produjo un conjunto de 20 transformaciones. Se aplicaron las transformaciones a las imágenes seleccionadas y se probaron los algoritmos sobre ellas, obteniendo como resultado para cada problema los 6 parámetros de una transformación afín. De estos parámetros se recuperaron el ángulo de rotación, factores de escala, cizallamiento y traslación. Dichos valores se obtuvieron mediante las siguientes relaciones [16]

$$e_x = \sqrt{\theta_1^2 + \theta_3^2}$$

$$e_y = \frac{\theta_1\theta_4 - \theta_2\theta_3}{\sqrt{\theta_1^2 + \theta_3^2}}$$

$$\sin \alpha = \frac{\theta_3}{e_x}$$

$$\cos \alpha = \frac{\theta_1}{e_x}$$

$$c_x = \frac{\theta_1\theta_2 + \theta_2\theta_4}{\sqrt{\theta_1^2 + \theta_3^2}}$$

$$t_1 = \theta_5, t_2 = \theta_6$$

Las relaciones anteriores son válidas bajo el supuesto de que la multiplicación de matrices se realizó en el orden que especifica la ecuación 4.4 y que no se realizó cizallamiento en el eje y .

Los valores encontrados se compararon con los reales.

4.3.1 Pruebas en Imágenes no Ruidosas

Se utilizaron las siguientes imágenes para realizar las comparaciones:

Un angiograma, con un tamaño de 128×128 píxeles. Una imagen de la princesa Diana, con un tamaño de 128×128 píxeles. Una imagen de resonancia magnética, con un tamaño de 256×256 píxeles.

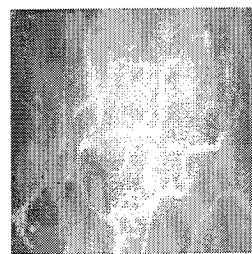


Figura 4-1: Imagen de un angiograma, utilizada para comparar los algoritmos de Gauss-Newton y de Estrategias Evolutivas.



Figura 4-2: Imagen de Diana, utilizada para comparar los algoritmos de Gauss-Newton y de Estrategias Evolutivas.

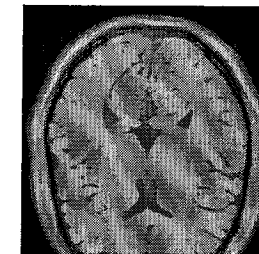


Figura 4-3: Imagen de una resonancia magnética, utilizada para comparar los algoritmos de Gauss-Newton y de Estrategias Evolutivas.

Los intervalos numéricos para los valores aleatorios de ángulos, factores de escala, cizallamiento y traslación, para cada conjunto de transformaciones se muestran en la tabla 4.1

Ambos métodos se ejecutaron bajo un esquema multiescala utilizando pirámides gaussianas de 4 niveles. Se utilizó una Estrategia Evolutiva (50 + 250). Se utilizó como punto inicial de ambos algoritmos la transformación identidad. Para el algoritmo Gauss-Newton, se permitió un máximo de 1000 iteraciones en cada nivel, pudiendo terminar en caso de que no se produjeran cambios significativos sobre el vector de parámetros. En el caso de estrategias evolutivas, se utilizaron 256 píxeles para evaluar la función de aptitud en cada nivel. Se permitió un máximo de 500 generaciones y el algoritmo terminaba al encontrar una diferencia de aptitud menor a 0.001 entre el mejor y el peor individuo de la población o al producir un individuo con aptitud mayor a 0.99.

| Parám. | 1er. Conj. | 2do. Conj. | 3er. Conj. | 4to. Conj. | 5to. Conj. |
|----------|------------------|------------------|------------------|------------------|------------------|
| θ | $\pm 10^\circ$ | $\pm 20^\circ$ | $\pm 30^\circ$ | $\pm 40^\circ$ | $\pm 50^\circ$ |
| e_x | 1 ± 0.1 | 1 ± 0.2 | 1 ± 0.3 | 1 ± 0.4 | 1 ± 0.5 |
| e_y | 1 ± 0.1 | 1 ± 0.2 | 1 ± 0.3 | 1 ± 0.4 | 1 ± 0.5 |
| c_x | ± 0.1 | ± 0.2 | ± 0.3 | ± 0.4 | ± 0.5 |
| t_1 | ± 10 pixeles | ± 20 pixeles | ± 30 pixeles | ± 40 pixeles | ± 50 pixeles |
| t_2 | ± 10 pixeles | ± 20 pixeles | ± 30 pixeles | ± 40 pixeles | ± 50 pixeles |

Tabla 4.1: Conjuntos de transformaciones

El vector ∇r_i utilizado por el método de Gauss-Newton, queda definido de la siguiente manera para el caso de transformaciones afines

$$\nabla r_i = \left[\frac{\delta r_i}{\delta \theta_1}, \dots, \frac{\delta r_i}{\delta \theta_6} \right] = [r_{x,i}x_i, r_{x,i}y_i, r_{x,i}, r_{y,i}x_i, r_{y,i}y_i, r_{y,i}]$$

Los valores de $r_{x,i}$ y de $r_{y,i}$ se calculan mediante diferencias finitas centrales de la siguiente manera

$$r_{x,i} = \frac{1}{2}[I_p(x_{i+1}, y_i) - I_p(x_{i-1}, y_i)]$$

$$r_{y,i} = \frac{1}{2}[I_p(x_i, y_{i+1}) - I_p(x_i, y_{i-1})]$$

$$I_p(x_i, y_i) = I_1[T(x_i, y_i)].$$

Los resultados obtenidos se presentan en las tablas ubicadas al final de este capítulo, en las cuales el símbolo Δ es utilizado para denotar el error promedio en algún parámetro. El tiempo reportado es el tiempo promedio del algoritmo sobre las pruebas exitosas. Los programas se corrieron en una PC, con procesador Xeon a 1.7 Ghz y 1 Gbyte de memoria RAM. Se tomó como éxito, una prueba en la que el error en el ángulo de rotación fuera menor a un grado, y el error en las componentes del vector de traslación fuera menor a un pixel.

| Conjunto | Algor. | $\Delta\theta$ | Δe_x | Δe_y | Δc_x | Δt_1 | Δt_2 | Tiempo | %éxito |
|------------|--------|----------------|--------------|--------------|--------------|--------------|--------------|--------|--------|
| 1er. Conj. | G-N | 0.0021° | 0.0000 | 0.0000 | 0.0000 | 0.0022 | 0.0033 | 5.9s | 65% |
| | E. Ev. | 0.0412° | 0.0032 | 0.0017 | 0.0028 | 0.2163 | 0.0938 | 15.2s | 100% |
| 2do. Conj. | G-N | 0.0001° | 0.0000 | 0.0000 | 0.0000 | 0.0007 | 0.0001 | 5.6s | 75% |
| | E. Ev. | 0.0752° | 0.0020 | 0.0020 | 0.0021 | 0.1484 | 0.1006 | 18.9s | 90% |
| 3er. Conj. | G-N | 0.0043° | 0.0001 | 0.0000 | 0.0002 | 0.0053 | 0.0083 | 6.2s | 65% |
| | E. Ev. | 0.0667° | 0.0036 | 0.0014 | 0.0045 | 0.1581 | 0.0741 | 21.0s | 100% |
| 4to. Conj. | G-N | 0.0004° | 0.0000 | 0.0000 | 0.0000 | 0.0009 | 0.0002 | 3.8s | 50% |
| | E. Ev. | 0.1732° | 0.0033 | 0.0020 | 0.0035 | 0.1264 | 0.0785 | 16.0s | 80% |
| 5to. Conj. | G-N | 0.0001° | 0.0000 | 0.0000 | 0.0000 | 0.0006 | 0.0001 | 2.05s | 25% |
| | E. Ev. | 0.0994° | 0.0030 | 0.0041 | 0.0037 | 0.1621 | 0.1828 | 14.65 | 70% |

Tabla 4.2: Resultados obtenidos con la imagen del angiograma

4.3.2 Pruebas en Imágenes Ruidosas

Para comparar los algoritmos se utilizó la misma imagen de resonancia magnética perturbada con ruido sobre la intensidad. Esta imagen se obtuvo del simulador del Instituto Neurológico de Montreal [25], en la Universidad de Mc Gill. La magnitud del ruido agregado fue del 9%, siendo este valor el máximo producido por el simulador. Se utilizó como imagen objetivo una transformación afín de la imagen ruidosa, y como imagen de referencia la imagen original sin ruido. Los resultados se muestran en las tablas ubicadas al final de este capítulo.

4.3.3 Conclusiones sobre las pruebas

En los resultados obtenidos podemos observar que el algoritmo de estrategias evolutivas presentó de manera general un mayor porcentaje de convergencia. El algoritmo de Gauss-Newton produjo resultados con errores en los parámetros menores que los generados por estrategias evolutivas. El tiempo de ejecución para el algoritmo de Gauss-Newton fue mucho menor que el tiempo de ejecución del algoritmo de estrategias evolutivas para las pruebas en imágenes sin ruido, sin embargo esta relación se invirtió en el caso de imágenes ruidosas.

| Conjunto | Algor. | $\Delta\theta$ | Δe_x | Δe_y | Δc_x | Δt_1 | Δt_2 | Tiempo | %éxito |
|------------|--------|----------------|--------------|--------------|--------------|--------------|--------------|--------|--------|
| 1er. Conj. | G-N | 0.0001° | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 5.25s | 85% |
| | E. Ev. | 0.0234° | 0.0009 | 0.0014 | 0.0022 | 0.0419 | 0.0635 | 14.5s | 100% |
| 2do. Conj. | G-N. | 0.0002° | 0.0000 | 0.0000 | 0.0000 | 0.0002 | 0.0004 | 4.90s | 60% |
| | E. Ev. | 0.0319° | 0.0006 | 0.0015 | 0.0023 | 0.0512 | 0.0327 | 13.65s | 100% |
| 3er. Conj. | G-N | 0.0003° | 0.0000 | 0.0001 | 0.0000 | 0.0016 | 0.0040 | 2.15s | 25% |
| | E. Ev. | 0.0226° | 0.0013 | 0.0003 | 0.0016 | 0.0409 | 0.0322 | 15.0s | 100% |
| 4to. Conj. | G-N | 0.0000° | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 4.80s | 40% |
| | E. Ev. | 0.0019° | 0.0000 | 0.0000 | 0.0000 | 0.0078 | 0.0052 | 15.8s | 70% |
| 5to. Conj. | G-N. | 0.0000° | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 1.70s | 30% |
| | E. Ev. | 0.0697° | 0.0015 | 0.0016 | 0.0019 | 0.0774 | 0.1152 | 10.3s | 70% |

Tabla 4.3: Resultados obtenidos con la imagen de Diana

| Conjunto | Algor. | $\Delta\theta$ | Δe_x | Δe_y | Δc_x | Δt_1 | Δt_2 | Tiempo | %éxito |
|------------|--------|----------------|--------------|--------------|--------------|--------------|--------------|--------|--------|
| 1er. Conj. | G-N | 0.0003° | 0.0000 | 0.0000 | 0.0000 | 0.0005 | 0.0005 | 16.8s | 100% |
| | E. Ev. | 0.0010° | 0.0000 | 0.0000 | 0.0000 | 0.0043 | 0.0035 | 19.8s | 100% |
| 2do. Conj. | G-N. | 0.0008° | 0.0000 | 0.0000 | 0.0000 | 0.0008 | 0.0009 | 13.75s | 90% |
| | E. Ev. | 0.0010° | 0.0000 | 0.0000 | 0.0000 | 0.0048 | 0.0037 | 17.9s | 90% |
| 3er. Conj. | G-N | 0.0000° | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 11.75s | 80% |
| | E. Ev. | 0.0011° | 0.0000 | 0.0000 | 0.0000 | 0.0052 | 0.0041 | 16.3s | 80% |
| 4to. Conj. | G-N | 0.0000° | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 3.90s | 55% |
| | E. Ev. | 0.0657° | 0.0012 | 0.0010 | 0.0020 | 0.0369 | 0.0443 | 12.0s | 80% |
| 5to. Conj. | G-N. | 0.0000° | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 7.4s | 45% |
| | E. Ev. | 0.0018° | 0.0000 | 0.0000 | 0.0000 | 0.0080 | 0.0062 | 18.6 | 80% |

Tabla 4.4: Resultados obtenidos con la imagen de resonancia

| Conjunto | Algor. | $\Delta\theta$ | Δe_x | Δe_y | Δc_x | Δt_1 | Δt_2 | Tiempo | %éxito |
|------------|--------|----------------|--------------|--------------|--------------|--------------|--------------|--------|--------|
| 1er. Conj. | G-N | 0.0095° | 0.0002 | 0.0002 | 0.0003 | 0.0360 | 0.0408 | 49.1s | 100% |
| | E. Ev. | 0.0794° | 0.0014 | 0.0010 | 0.0023 | 0.2250 | 0.2402 | 16.55s | 100% |
| 2do. Conj. | G-N. | 0.0142° | 0.0003 | 0.0003 | 0.0004 | 0.0543 | 0.0399 | 36.25s | 90% |
| | E. Ev. | 0.0618° | 0.0019 | 0.0010 | 0.0030 | 0.2428 | 0.1976 | 14.5s | 90% |
| 3er. Conj. | G-N | 0.0095° | 0.0003 | 0.0003 | 0.0005 | 0.0562 | 0.0437 | 27.3s | 75% |
| | E. Ev. | 0.0588° | 0.0009 | 0.0008 | 0.0021 | 0.1279 | 0.1406 | 11.5s | 70% |
| 4to. Conj. | G-N | 0.0073° | 0.0001 | 0.0000 | 0.0003 | 0.0274 | 0.0191 | 21.85s | 45% |
| | E. Ev. | 0.0613° | 0.0007 | 0.0006 | 0.0015 | 0.0922 | 0.1252 | 10.7s | 65% |
| 5to. Conj. | G-N. | 0.0066° | 0.0002 | 0.0002 | 0.0008 | 0.0210 | 0.0245 | 17.55s | 45% |
| | E. Ev. | 0.0789° | 0.0015 | 0.0007 | 0.0026 | 0.1494 | 0.2006 | 9.2s | 55% |

Tabla 4.5: Resultados obtenidos con la imagen de resonancia con ruido

Capítulo 5

Registro de Imágenes Basado en Información Mutua

5.1 Antecedentes

Cuando la función de transferencia de tonos Φ , entre la imagen de referencia y la imagen objetivo es distinta a la identidad, las técnicas de registro presentadas en el capítulo anterior no pueden ser aplicadas, a menos que se tenga conocimiento explícito de Φ .

El registro de imágenes mediante la Maximización de la Información Mutua [21], es una técnica que ha generado grandes expectativas recientemente ya que basa su aplicación en que podemos obtener la mayor información sobre Φ cuando las imágenes objetivo y de prueba se encuentran perfectamente alineadas.

La información mutua entre dos variables aleatorias se define como

$$IM(x, y) = H(x) + H(y) - H(x, y) \quad (5.1)$$

$H(x)$, se denomina función de *entropía* y se define de la siguiente manera

$$H(x) = - \int p(x) \log p(x) dx \quad (5.2)$$

$H(\mathbf{x}, \mathbf{y})$ es la *entropía conjunta* de las variables \mathbf{x}, \mathbf{y} y es igual a

$$H(\mathbf{x}, \mathbf{y}) = - \int p(\mathbf{x}, \mathbf{y}) \log p(\mathbf{x}, \mathbf{y}) d\mathbf{x}d\mathbf{y} \quad (5.3)$$

La entropía mide la incertidumbre sobre el estado de una variable, de manera que su valor es máximo cuando se aplica a una variable con una distribución de probabilidad uniforme. De la misma forma, la entropía conjunta es máxima cuando las variables \mathbf{x}, \mathbf{y} son independientes, en tal caso se comprueba fácilmente que

$$H(\mathbf{x}, \mathbf{y}) = H(\mathbf{x}) + H(\mathbf{y})$$

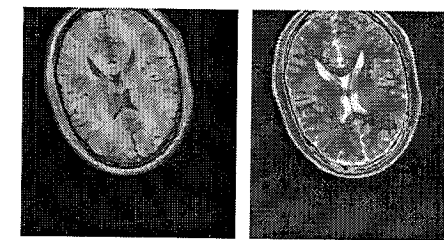
Bajo este planteamiento podemos entender que la información mutua será mínima e igual a cero, cuando \mathbf{x} e \mathbf{y} sean independientes y tendrá valores mayores mientras más relacionadas estén.

Podemos identificar la intensidad en un punto cualquiera de la imagen objetivo, $I_2(\mathbf{x})$, y su valor correspondiente en la imagen de prueba, $I_1[T(\mathbf{x})]$, como dos variables aleatorias y replantear la ecuación 5.1 para definir la información mutua entre ambas imágenes

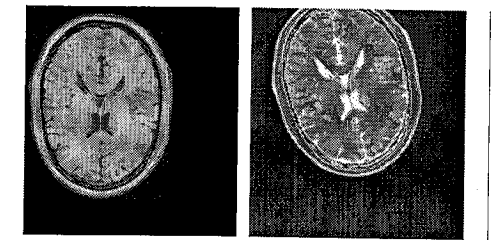
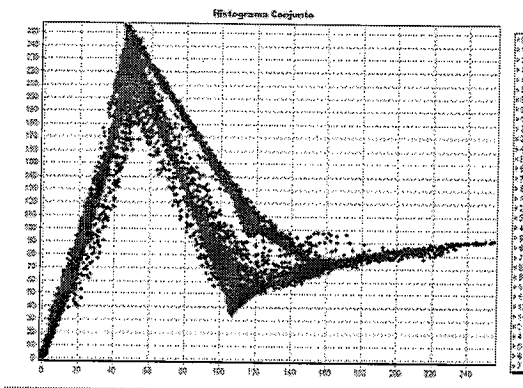
$$IM\{I_1[T(\mathbf{x})], I_2(\mathbf{x})\} = H\{I_1[T(\mathbf{x})]\} + H\{I_2(\mathbf{x})\} - H\{I_1[T(\mathbf{x})], I_2(\mathbf{x})\} \quad (5.4)$$

Cuando ambas imágenes se encuentran perfectamente alineadas, la correlación entre las intensidades se presenta a través de la función de transferencia de tonos, a diferencia de cuando no lo están, en cuyo caso las intensidades se comportan como variables independientes, como se observa en la figura 5-1, en la cual también puede observarse que Φ no siempre es una función estrictamente hablando.

Así pues, el registro de imágenes puede llevarse a cabo encontrando los parámetros de la transformación que maximicen la información entre las imágenes de prueba y objetivo.



Imágenes registradas



Imágenes no registradas

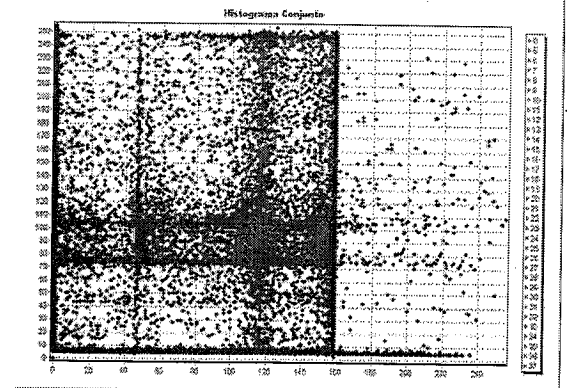


Figura 5-1: En esta figura se muestran los histogramas conjuntos de pares de imágenes alineadas y no alineadas. El histograma fue construido tomando en cuenta los tonos de gris en los pixeles correspondientes de cada par de imágenes. En cada eje se representan los valores numéricos de los tonos de gris de cada imagen.

5.1.1 Ventanas de Parzen

Un inconveniente que surge al utilizar la ecuación 5.1 es la necesidad de conocer las funciones de distribución de las variables aleatorias, lo cual no siempre es posible.

Este problema puede ser superado mediante la utilización de las Ventanas de Parzen [24], las cuales son funciones que se centran sobre cada una de las muestras de un conjunto generado a través de la distribución que se desea modelar, de manera que la función de probabilidad puede escribirse

$$p(\mathbf{x}) = \frac{1}{N_A} \sum_{\mathbf{x}_i \in A} \Psi(\mathbf{x} - \mathbf{x}_i)$$

N_A es el número de muestras utilizadas para modelar la distribución.

Ψ , es una función que cumple con $\int \Psi(\mathbf{x})d\mathbf{x} = 1$, $\Psi(\mathbf{x}) \geq 0$.

Las funciones gaussianas son utilizadas ampliamente como ventanas de Parzen, de manera que podemos usar

$$\Psi(\mathbf{x}) = G_C(\mathbf{x}) = \frac{1}{(2\pi)^{\frac{n}{2}} |C|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}\mathbf{x}^T C^{-1}\mathbf{x}\right)$$

C , es una matriz de covarianza.

n , es la dimensión de \mathbf{x} .

La elección de la matriz de covarianza es un punto importante a considerar, como puede observarse en la figura 5-2, en la cual se muestran los modelos obtenidos para distintos valores de varianza.

5.2 Registro por Maximización Estocástica de la Información Mutua

En [22], se propone la maximización de la Información Mutua a través de un proceso de ascenso de gradiente estocástico con paso fijo. Para la evaluación de las ecuaciones 5.2 y 5.3 utiliza ventanas de Parzen gaussianas centradas en un conjunto de observaciones de la variable aleatoria y la integral es sustituida por un simple promedio aritmético, con lo que la ecuación 5.2 se reescribe como

$$H(\mathbf{x}) = -\frac{1}{N_B} \sum_{\mathbf{x}_i \in B} \ln \frac{1}{N_A} \sum_{\mathbf{x}_j \in A} G_C(\mathbf{x}_i - \mathbf{x}_j) \quad (5.5)$$

En esta aproximación se usan 2 conjuntos de muestreo sobre la variable aleatoria x , los cuales se denominan A y B , el primero se utiliza para la modelación de la función de probabilidad y el segundo para evaluar la entropía.

La maximización de la ecuación 5.4 requiere del cálculo del gradiente con respecto a

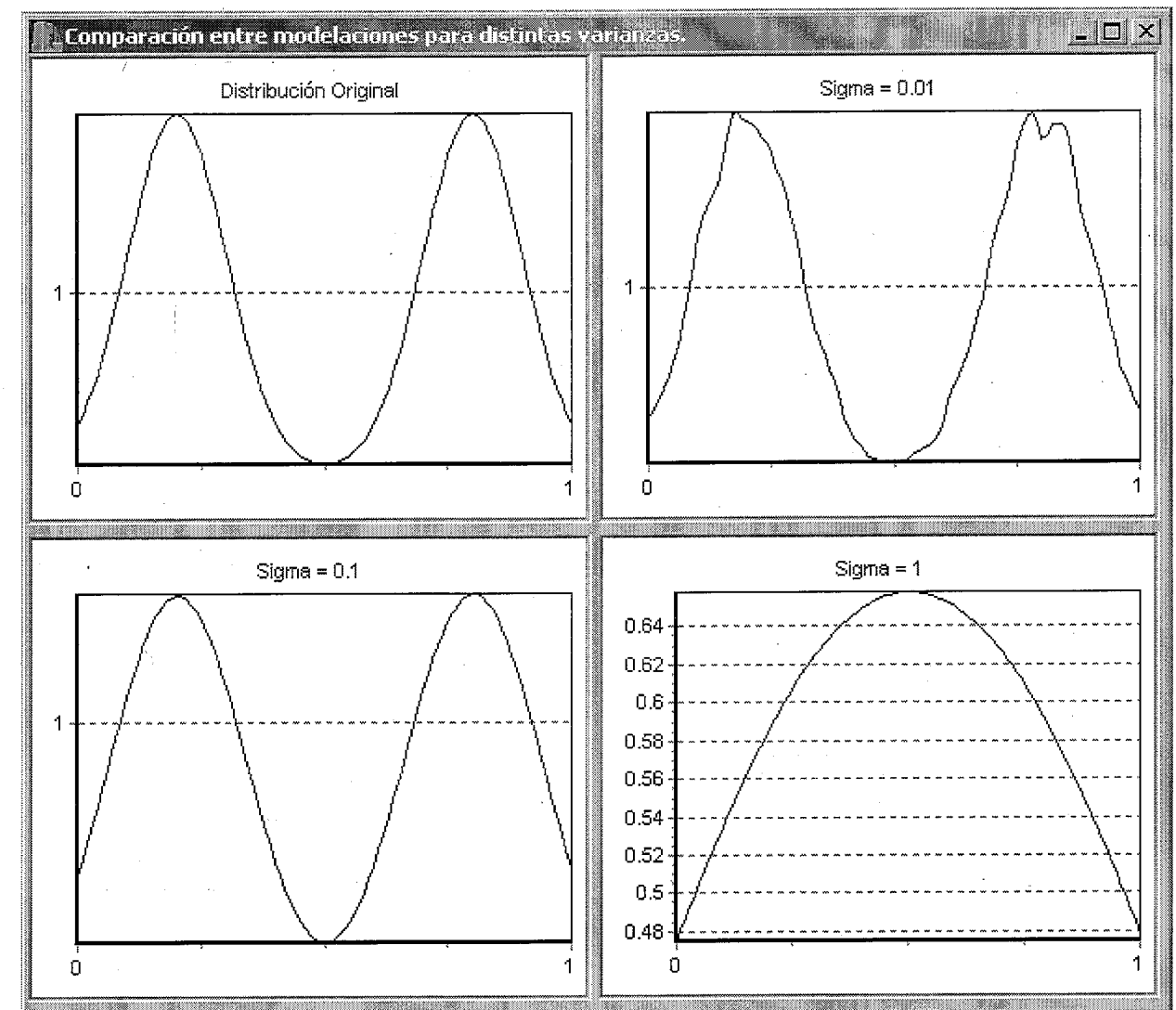


Figura 5-2: Modelos obtenidos utilizando gaussianas con diferentes varianzas como ventanas de Parzen. Se utilizaron 100 puntos de muestreo de la distribución de la esquina superior izquierda.

los parámetros de la transformación $T(\theta)$

$$\nabla MI = \nabla H(I_1(T(\theta))) - \nabla H(I_1(T(\theta)), I_2)$$

$$\nabla = \left\{ \frac{\delta}{\delta\theta_1}, \dots, \frac{\delta}{\delta\theta_m} \right\}$$

El gradiente de la imagen objetivo no aparece debido a que ésta no depende de los parámetros de la transformación.

El cálculo del gradiente de la entropía puede realizarse mediante la aproximación 5.5

$$\nabla H(I_1(T(\theta))) = \frac{1}{N_B} \sum_{x_i \in B} \sum_{x_j \in A} W_v(v_i, v_j) (v_i - v_j)^T C_v^{-1} \nabla(v_i - v_j) \quad (5.6)$$

$$v_i = I_1(T(\theta, \mathbf{x}_i)), v_j = I_1(T(\theta, \mathbf{x}_j)), v_k = I_1(T(\theta, \mathbf{x}_k))$$

$$W_v = \frac{G_{C_v}(v_i - v_j)}{\sum_{x_k \in A} G_{C_v}(v_i - v_k)}$$

A y B , son conjuntos de muestreo sobre las coordenadas de la imagen I_2 .

El cálculo del gradiente de la entropía conjunta requiere de la construcción del vector $\mathbf{w} = [I_1(T(\theta, \mathbf{x})), I_2(\mathbf{x})]$, con lo que la ecuación 5.6 puede reescribirse como

$$\nabla MI = \frac{1}{N_B} \sum_{x_i \in B} \sum_{x_j \in A} (v_i - v_j)^T (W_v(v_i, v_j) C_v^{-1} - W_w(\mathbf{w}_i, \mathbf{w}_j) C_w^{-1}) \nabla(v_i - v_j)$$

$$W_v = \frac{G_{C_v}(v_i - v_j)}{\sum_{x_k \in A} G_{C_v}(v_i - v_k)}$$

$$W_w = \frac{G_{C_w}(\mathbf{w}_i - \mathbf{w}_j)}{\sum_{x_k \in A} G_{C_w}(\mathbf{w}_i - \mathbf{w}_k)}$$

El proceso de registro mediante maximización de la información mutua por ascenso de gradiente estocástico se resume en el siguiente esquema

- 1.- Iniciar parámetros θ^0 .
- 2.- Iniciar contador de iteraciones $t = 0$.
- 3.- Repetir
 - generar conjunto A

generar conjunto B

actualizar $\theta^{t+1} = \theta^t + \lambda \nabla MI(I_1(T(\theta, \mathbf{x})), I_2(\mathbf{x}))$.

$t = t + 1$.

4.- Fin.

El proceso se denomina estocástico ya que la aleatoriedad de los conjuntos A y B , introduce ruido numérico en el cálculo del vector gradiente, lo cual debe permitir el escape de óptimos locales. Sin embargo, al mismo tiempo resulta difícil encontrar un buen criterio de paro para el algoritmo, pues el vector gradiente difícilmente será igual a cero en el óptimo, por lo que generalmente el proceso iterativo se repite un número fijo de veces.

5.3 Registro por Maximización de la Información Mutua mediante Estrategias Evolutivas

En esta sección se presenta otra propuesta para el registro de imágenes mediante la utilización de estrategias evolutivas.

La codificación, recombinación y mutación se realizan según lo establecido en la sección 4.2.3. La función de aptitud es la información mutua entre la imagen objetivo y la imagen de prueba que representa cada individuo

$$\Phi(\mathbf{a}) = IM\{I_1[T(\theta, \mathbf{x})], I_2(\mathbf{x})\} = H\{I_1[T(\theta, \mathbf{x})]\} + H[I_2(\mathbf{x})] - H\{I_1[T(\theta, \mathbf{x})], I_2(\mathbf{x})\}$$

θ , es el vector de variables objetivo del individuo \mathbf{a} .

La función de probabilidad se modela con ventanas de Parzen gaussianas, y la entropía se evalúa mediante la relación 5.5, utilizando dos conjuntos de puntos tomados aleatoriamente sobre las coordenadas de la imagen objetivo.

Los conjuntos de muestreo son renovados cada generación, por lo cual la aptitud de cada individuo dependerá del conjunto de muestreo utilizado, y la selección debe tomar en cuenta que la aptitud de los padres y la de los hijos son calculadas mediante conjuntos diferentes. Lo anterior obliga a recalcular la aptitud de los padres en cada generación.

5.4 Comparación entre Estrategias Evolutivas y Ascenso de Gradiente Estocástico

La comparación entre las estrategias de optimización propuestas en este capítulo, se realizó a través del registro de imágenes bidimensionales de resonancia magnética de tipo T1 y T2, obtenidas del simulador de la Universidad de Mc. Gill. Ambas imágenes representan una misma estructura cerebral y se escaló el rango dinámico de las imágenes al intervalo $[0, 1]$. La imagen de tipo T1, fig. 5-3, fue utilizada como imagen de referencia, y la imagen T2, fig 5-4, perturbada con un 9% de ruido y un 40% de inhomogeneidad sobre tonos de gris, fue utilizada como imagen objetivo después de aplicársele una transformación rígida.

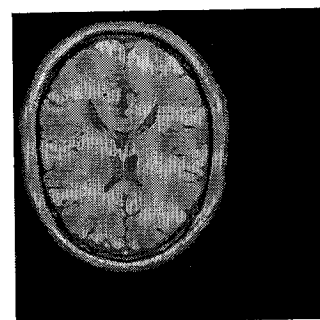


Figura 5-3: Imagen de resonancia magnética de modalidad T1 utilizada como imagen de referencia.

Se generaron 4 conjuntos de 10 transformaciones rígidas, con valores de ángulos de rotación y vector de traslación escogidos aleatoria y uniformemente sobre intervalos nu-

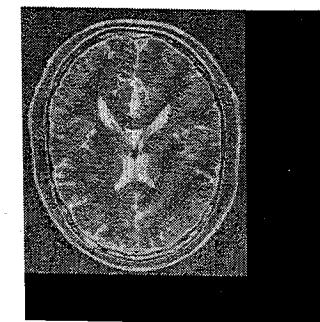


Figura 5-4: Imagen de resonancia magnética de modalidad T2, a la cual se aplicó una transformación rígida y fue utilizada como imagen objetivo.

| Parám. | 1er. Conj. | 2do. Conj. | 3er. Conj. | 4to. Conj.. |
|----------|----------------|----------------|----------------|----------------|
| θ | $\pm 10^\circ$ | $\pm 20^\circ$ | $\pm 30^\circ$ | $\pm 45^\circ$ |
| t_1 | ± 10 | ± 10 | ± 15 | ± 50 |
| t_2 | ± 10 | ± 10 | ± 15 | ± 50 |

Tabla 5.1: Conjuntos de transformaciones aplicadas

méricos que fueron incrementándose gradualmente. Los intervalos se muestran en la tabla 5.1

Ambos algoritmos se utilizaron aplicados a registro afín, por lo que el cálculo del vector $\nabla(v_i - v_j)$, necesario para el método de ascenso de gradiente estocástico, se llevó a cabo utilizando las relaciones descritas en la sección 4.3.1.

Se recuperó el ángulo de rotación aplicando el método de descomposición en valores singulares, *SVD* [23], el cual encuentra la factorización de la matriz simétrica más parecida a la matriz A de la ecuación 2.2, mediante la siguiente relación

$$A \approx UWV$$

en la que las matrices U y V , son matrices ortogonales y W es una matriz diagonal.

La matriz de rotación recuperada es el producto de U y V , de la cual se recuperó

| Conjunto | Algoritmo | $\Delta\theta$ | Δt_1 | Δt_2 | Tiempo prom. | % de éxito |
|---------------|------------|----------------|--------------|--------------|--------------|------------|
| 1er. Conjunto | Grad. Est. | 0.0341 | 0.2342 | 0.2413 | 344 s | 90% |
| | Est. Ev. | 0.1152 | 0.4383 | 0.4069 | 664 s | 80% |
| 2do. Conjunto | Grad. Est. | 0.0259 | 0.2620 | 0.2236 | 341 s | 70% |
| | Est. Ev. | 0.1618 | 0.3878 | 0.4829 | 1,144 s | 90% |
| 3er. Conjunto | Grad. Est. | 0.0503 | 0.2406 | 0.2178 | 334.5 s | 40% |
| | Est. Ev. | 0.3128 | 0.4399 | 0.3059 | 935 s | 80% |
| 4to. Conjunto | Grad. Est. | 0.0457 | 0.2933 | 0.1604 | 338 s | 20% |
| | Est. Ev. | 0.1286 | 0.3883 | 0.1418 | 893 s | 40% |

Tabla 5.2: Resumen de resultados obtenidos

el ángulo de rotación de la transformación afín definida por A . Los valores encontrados se compararon con los reales y se consideró un éxito cuando las diferencias entre los ángulos fue menor a un grado y entre los vectores de traslación menor a un pixel en cada componente.

Los algoritmos se ejecutaron bajo un esquema multiescala, utilizando una pirámide gaussiana de 3 niveles. En el caso de ascenso de gradiente estocástico se utilizaron 10,000 repeticiones en cada nivel y un valor de 0.1 para el tamaño de paso, $\lambda = 0.1$. Se aplicó una Estrategia Evolutiva (50 + 250), ejecutando un máximo de 500 generaciones en cada nivel, pudiendo detenerse al existir una diferencia menor a 0.001 entre las aptitudes del mejor y del peor individuo.

Se utilizaron desviaciones estándar fijas ($\sigma = 0.1$), para las gaussianas con las que se modeló la función de probabilidad utilizada en la entropía de la imagen de prueba, y como matriz de covarianza para la función de probabilidad de la entropía conjunta la

$$\text{matriz diagonal} \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}.$$

Los resultados se presentan en la tabla 5.2

Capítulo 6

Conclusiones y Trabajo Futuro

En este trabajo han sido presentados dos algoritmos de registro paramétrico basados en la utilización de estrategias evolutivas como técnica de optimización, los cuales son aplicados tanto a registro monomodal como al multimodal.

De las pruebas realizadas, podemos afirmar que dichos algoritmos tuvieron un comportamiento robusto ante la presencia de ruido y de grandes transformaciones, sobre diferentes imágenes.

De manera general, los algoritmos propuestos tuvieron un mejor desempeño que los tradicionales, basados en el porcentaje de éxito sobre las comparaciones realizadas, sin embargo el tiempo promedio por registro fue muy superior al utilizado por los algoritmos tradicionales, algo similar podemos afirmar sobre los errores promedio. Por lo anterior sería interesante construir una metodología híbrida, ya que podrían combinarse la robustez de las estrategias evolutivas con la gran precisión y rapidez de los métodos tradicionales.

La utilización de otras funciones de aptitud como la Información Mutua Normalizada o la función de Correlación, es otro punto interesante que será revisado en un futuro próximo.

La generalización de los algoritmos propuestos hacia registro sobre imágenes tridimensionales es directa, sin embargo, se deberá tener en cuenta la optimización de los

tamaños de las poblaciones y del número de puntos utilizados para la evaluación de la función de aptitud, sobre todo en el registro monomodal.

Bibliografía

- [1] John H. Holland. *Adaptation in natural and artificial systems*. The University of Michigan Press. Ann Arbor, 1975.
- [2] Kenneth De Jong. *An analysis of the behavior of a class of genetic adaptive systems*. Ph.D. Thesis. University of Michigan, 1975.
- [3] Lawrence J. Fogel, A.J. Owens, and M.J. Walsh. *Artificial intelligence through simulated evolution*. Wiley, New York, 1966.
- [4] David B. Fogel. *Evolving artificial intelligence*. Ph.D. Thesis, University of California, San Diego, 1992.
- [5] Ingo Rechenberg. *Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. FrommannHolzboog, Stuttgart, 1973.
- [6] Hans-Paul Schwefel. *Evolutionsstrategie und numerische Optimierung*. Dr.-Ing. Thesis, Technical. University of Berlin, Department of Process Engineering, 1975.
- [7] Hans-Paul Schwefel. *On the evolution of evolutionary computation*. *Computational Intelligence – Imitating Life*. pp. 116–124. IEEE Press, Piscataway NJ. 1994.0-7803-1104-3.
- [8] Hans-Paul Schwefel and Günter Rudolph. *Contemporary evolution strategies*. *Advances in Artificial Life, Proc. Third European Conf. Artificial Life (ECAL'95)*. Springer, Berlin 1995.

- [9] Jean-Jose Jacq and Christian Roux. Registration of non-segmented images using a genetic algorithm. First International Conference on Computer Vision, Virtual Reality and Robotics in Medicine. Springer. France, 1995.
- [10] Prachya Chalermwat and Tarek El-Ghazawi and Jacqueline LeMoigne. GA-Based Parallel Image Registration on Parallel Clusters. IPPS/SPDP Workshops. 257-265. 1999.
- [11] Lisa Gottesfeld Brown. A survey of image registration techniques. ACM Computing Surveys, 24(4):325-376, December 1992. <http://citeseer.nj.nec.com/gottesfeldbrown92survey.html>.
- [12] J. B. A. Maintz and M. A. Viergever, "A survey of medical image registration," Medical Image Analysis 2(1), pp. 1-36, 1998. (Web version: http://www.oup.co.uk/media/j/hdb/Volume_02/Issue_01/). <http://citeseer.nj.nec.com/maintz98survey.html>
- [13] Tony Lindeberg. Scale space theory in computer vision. Kluwer academic publishers. Netherlands, 1994.
- [14] J. L. Marroquín. Apuntes de clase "Visión Computacional II". México 2001.
- [15] M. Unser, "Splines: A Perfect Fit for Signal and Image Processing," IEEE Signal Processing Magazine, vol. 16, no. 6, pp. 22-38, November 1999.
- [16] K. Voss and H. Suesse. Comunicaciones Técnicas del CIMAT, 05.06.2001, I-01-06 (CC).
- [17] R. Szeliski and J. Coughlan. Spline-based image registration. International Journal of Computer Vision, 22(3):199-218, March/April 1997
- [18] Thomas Back, Frank Hoffmeister, and Hans-Paul Schwefel. A survey of evolution strategies. In Lashon B. Belew, Richard K.; Booker, editor, Proceedings of the 4th

- International Conference on Genetic Algorithms, pages 2-9, San Diego, CA, July 1991. Morgan Kaufmann. <http://citeseer.nj.nec.com/back91survey.html>
- [19] Thomas Bäck. Evolutionary Algorithms in Theory and Practice. Oxford University Press. New York, 1996.
- [20] Jorge Nocedal and Stephen Wright. Numerical Optimization. Springer-Verlag, New York. 1999.
- [21] P.A. Viola. Alignment by Maximization of Mutual Information. PhD thesis, Massachusetts Institute of Technology (MIT), Cambridge, Massachusetts, March 1995. <http://citeseer.nj.nec.com/viola95alignment.html>
- [22] Wells, W. M., Viola, P., Atsumi, H., Nakajima, S., and Kikinis, R. (1996). Multimodal volume registration by maximization of mutual information. Medical Image Analysis, 1(1):35-51. <http://citeseer.nj.nec.com/wells96multimodal.html>.
- [23] William H. Press, Saul A. Teukolsky, William T. Vetterling and Brian P. Flannery. Numerical recipes in C. Cambridge University Press. 1992.
- [24] R. O. Duda and P. E. Hart. Pattern Classification and Scene Analysis, John Wiley and Sons, 1973.
- [25] <http://www.bic.mni.mcgill.ca/brainweb/>
- [26] C.A. Cocosco, V. Kollokian, R.K.-S. Kwan, A.C. Evans : "BrainWeb: Online Interface to a 3D MRI Simulated Brain Database". NeuroImage, vol.5, no.4, part 2/4, S425, 1997. Proceedings of 3-rd International Conference on Functional Mapping of the Human Brain, Copenhagen, May 1997.
- [27] R.K.-S. Kwan, A.C. Evans, G.B. Pike : "MRI simulation-based evaluation of image-processing and classification methods". IEEE Transactions on Medical Imaging. 18(11):1085-97. Nov1999.

- [28] R.K.-S. Kwan, A.C. Evans, G.B. Pike : "An Extensible MRI Simulator for Post-Processing Evaluation". Visualization in Biomedical Computing (VBC'96). Lecture Notes in Computer Science. vol. 1131. Springer-Verlag, 1996. 135-140.
- [29] D.L. Collins, A.P. Zijdenbos, V. Kollokian, J.G. Sled, N.J. Kabani, C.J. Holmes, A.C. Evans : "Design and Construction of a Realistic Digital Brain Phantom". IEEE Transactions on Medical Imaging, vol.17, No.3, p.463-468, June 1998.

| FECHA | DEVOLUCION |
|-------|------------|
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |

**CENTRO DE INVESTIGACION
EN MATEMATICAS, A.C.**

CONTROL DE PRESTAMO
BIBLIOTECA

•019304

•TE 140 Ej.4

•GOMEZ GARCIA, HECTOR FERN

•ESTRATEGIAS EVOLUTIVAS AP

Nombre del Usuario

TE
140

Bib. CIMAT
Ej.4

