Análisis e Implementación de algoritmos para el mejoramiento de Imágenes de Radiografías en busca de Detalles

Guzmán Real Carlos Julio[1], González Linch Erick Ermory[2], Chávez Patricia[3]
Facultad de Ingeniería Eléctrica y Computación (FIEC)
Escuela Superior Politécnica del Litoral (ESPOL)
Campus Gustavo Galindo, Km 30.5 Vía Perimetral
Apartado 09-01-5863 Guayaquil, Ecuador
carlos_julioguzman@hotmail.com[1], doble_e5@hotmail.com[2], pchavez@espol.edu.ec[3]

Resumen

El siguiente articulo reúne una serie de herramientas implementadas en Matlab con el fin de mejorar las imágenes de radiografía para confirmar o descartar fisuras o fracturas que pudieran ser imperceptibles al ojo humano debido a su pobre capacidad de no poder diferenciar mas allá de 80 valores en la escala de grises. El tema de la calidad de las radiografías es un asunto de gran importancia para la medicina encargada de estudiar los traumas con los que se ve afectado el cuerpo humano, su aguda visualización es de vital importancia para poder realizar un correcto diagnostico. Este informe analiza y prueba brevemente algunos algoritmos con el fin de acentuar y esclarecer rasgos imperceptibles de ciertas radiografías para ayudar en el diagnostico de las mismas, utilizando el procesamiento digital de imágenes. Se desarrollará una aplicación en Matlab, principalmente basada en la aplicación de mascaras y filtros para mejorar la detección de bordes, además de otras herramientas como ajuste de curva las cuales ayudaran a mejorar la imágenes de radiografía, se espera que esta aplicación sea de fácil uso para cualquier usuario, y este sea capaz de obtener los resultados deseados.

Palabras Claves: *Radiografía, filtros, ajuste de curva, fisuras, mascaras, procesamiento digital de imágenes*

Abstract

The following article has a number of tools implemented in Matlab in order to improve x-ray images to confirm or discard fractures and fissures that may be imperceptible to the human eye due to its poor ability of not being able to differentiate beyond 80 values in the grayscale. The x-ray image quality has a great importance in the medicine field related to trauma analysis, which affect the human body, its sharpening quality is vital to decide a right diagnosis. This report will analyze and test briefly, some algorithms in order to clarify and emphasize certain hidden traits imperceptibles in x-ray images using digital image processing. Will be develop an application in Matlab, mainly based on the use of masks for filtering and edge detection and other tools such as curve adjustment, all of these algorithms will help in the radiographies images improvement, this program is easy to use for any user, and the doctor will obtain the desired results.

Keywords: *X-Ray images, filters, mask, curve adjustment, digital image processing, fissures*

1. Introducción

El procesamiento digital de imágenes es el conjunto de técnicas que se aplican a las imágenes digitales con el objetivo de mejorar la calidad o facilitar la búsqueda de información en dichas imágenes.

De manera específica nuestro objetivo es realizar un procesamiento que permita identificar detalles, prácticamente imperceptibles a simple vista, lo que sin duda será de gran ayuda en el área de la medicina.

Los filtros y ajustes son las principales herramientas para lograr nuestro objetivo, ya que mediante su uso podremos detectar bordes, modificar es aspecto de las imágenes o sencillamente eliminar ruido de las mismas, nuestro proyecto los utilizará para mejorar los detalles en los bordes de los huesos tomados en radiografías, con fin de proporcionar al doctor una mejor imagen que lo ayude en su diagnostico.

Para el desarrollo del proyecto utilizaremos MATLAB V7.7.0.471 (R2008b) ya que es un programa interactivo para computación numérica y visualización de datos. Es ampliamente usado por Ingenieros de Control en el análisis y diseño, además tiene una extraordinaria versatilidad y capacidad para resolver problemas en matemática aplicada, física, química, ingeniería, finanzas y muchas otras aplicaciones. Está basado en un sofisticado software de matrices para el análisis de sistemas de ecuaciones. Permite resolver complicados problemas numéricos sin necesidad de escribir un programa.

2. Filtrado de Imágenes

Los filtros son operaciones que se aplican a los píxeles de una imagen digital para mejorarla, resaltar cierta información o conseguir un efecto especial.

El filtrado digital de imágenes es una operación de convolución de la imagen original con la función del filtro, la operación de convolución entre las funciones $f_1(t)$ y $f_2(t)$ se define de la siguiente manera:

$$f(t) = f_1(t) * f_2(t) = \int_{-\infty}^{\infty} f_1(x) f_2(t-x)dx$$

2.1. Dominio espacial y de la frecuencia

Las técnicas de procesamiento de punto o de convolución que operan sobre la imagen en el dominio espacial calculan los valores sobre los propios valores de los píxeles, a diferencia de los del dominio de la frecuencia que la traducen primero a un mapa de frecuencias mediante una transformada de tipo Fourier.

Nuestra acción será sobre el filtrado espacial, el mismo que se realiza definiendo un entorno con los vecinos de un píxel central. Este entorno se denomina ventana, máscara o matriz de convolución, y suele ser cuadrado o rectangular, aunque es posible definir otras formas rómbicas, circulares, etc.

A cada posición en la ventana se le asigna un peso o participación en el cálculo que dará el nuevo valor para el píxel central. Entonces, se va desplazando la máscara, centrándola en cada uno de los píxeles de la imagen. Al recorrido completo es a lo que denominamos filtrado, y los sucesivos resultados, siempre a partir de los valores originales, forman la nueva imagen. Es obvio que en los bordes y esquinas la ecuación varía, ya que parte de la máscara se pierde ([1]).

Es importante saber que los ejes o bordes de una imagen se encuentran en zonas donde el nivel de intensidad cambia bruscamente, cuanto más rápido se produce el cambio de intensidad, el borde es más fuerte. Un buen proceso de detección de bordes facilita la elaboración de las fronteras de objetos con lo que, el reconocimiento de objetos se simplifica.

2.2. Filtros Utilizados

2.2.1. Unsharp. El filtro Unsharp es un simple operador que busca nitidez, y es una de las funciones dentro de 'fspecial' de MATLAB, su nombre se infiere debido al hecho que aumenta los bordes además de algunos componentes de altas frecuencias de la imagen. Mediante un procedimiento resta una versión alisada de la imagen original, lo que aumenta considerablemente la nitidez del resultado ([2]).

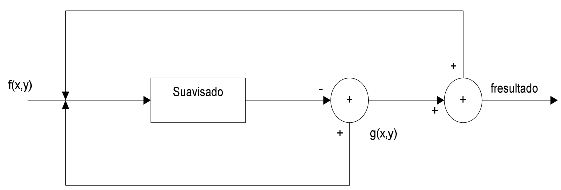


Figura 1. Filtro Unsharp.

$$f_{\mathrm{Resultado}}(x,y) = f(x,y) + g(x,y)$$

[1] *www.quesabesde.com Paulo Porta, Técnicas de Filtrado http://www.quesabesde.com/camdig/articulos.asp?articulo=137*

[2] *http://homepages.inf.ed.ac.uk/rbf/HIPR2/unsharp.htm*

Tabla 1. Mascara del Filtro Unsharp

-0.1667	-0.6667	-0.1667
-0.6667	4.3333	-0.6667
-0.1667	-0.6667	-0.1667

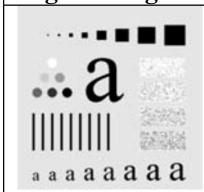
Figura Original	Figura Unsharp
	

Figura 2. Comparación Unsharp.

2.2.2. Filtro Emboss. El filtro Emboss, conocido como de "Repujado" hace que una imagen aparezca estampada debido a la supresión del color en la imagen y marcando sus bordes con negro ([3])

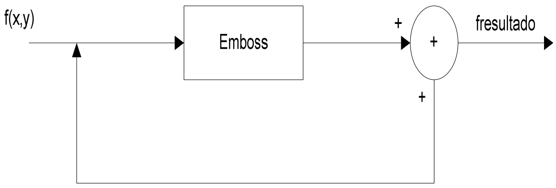


Figura 3. Filtro Emboss.

$$f_{\mathrm{Re}\,sultado}(x, y) = f(x, y) + f(x, y) * H(x, y)$$

Tabla 2. Mascara del Filtro Emboss

-2	-1	0
-1	1	1
0	1	2

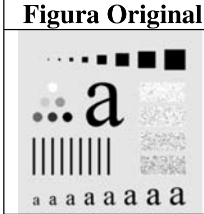
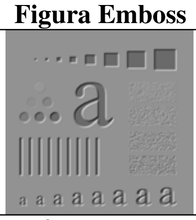
Figura Original	Figura Emboss
	

Figura 4. Comparación Emboss.

2.2.3. Filtro de Realce 1 y 2. Dos matrices son utilizadas para filtrar la imagen original, y de esta manera realizar el realce de la misma. Se puede ver cómo estas máscaras tienen un efecto de afilado de los bordes y mejora de los detalles borrosos sin que cambie la apariencia global de la imagen ([4]).

Figura 5. Filtro de Realce.

$$f_{\mathrm{Re}\,sultado}(x, y) = f(x, y) + f(x, y) * H(x, y)$$

Tabla 3. Mascara del Filtro de Realce 1

-1	-1	-1
-1	9	-1
-1	-1	-1

Tabla 4. Mascara del Filtro de Realce 2

-0.1429	-0.1176	-0.1429
-0.1176	2.7143	-0.1176
-0.1429	-0.1176	-0.1429

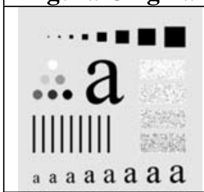
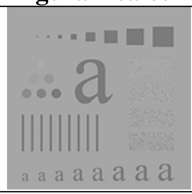
Figura Original	Figura Realce 1
	

Figura 6. Comparación Filtro de Realce 1.

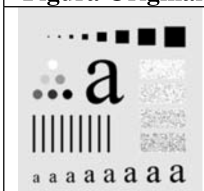
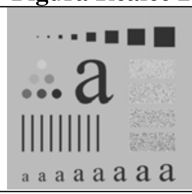
Figura Original	Figura Realce 2
	

Figura 7. Comparación Filtro de Realce 2.

2.2.4. Roberts, Prewitt y Sobel. Podemos realizar una clasificación general de los bordes según sea su dirección en:

- *Bordes verticales*, cuando píxeles conectados verticalmente tienen valores diferentes respecto de los anteriores o posteriores.
- *Bordes horizontales*, cuando tenemos píxeles conectados horizontalmente, y estos tienen distintos valores respecto de los anteriores o posteriores.

---

[3] *http://docs.gimp.org/en/plug-in-emboss.html*

[4] *http://www.des.udc.es/~adriana/TercerCiclo/CursoImagen/curso/web/Filtrado_Espacial_Real.html*

- *Bordes oblicuos,* cuando tenemos una combinación de las componentes horizontales y verticales (5).

Los operadores de **Prewitt**, **Sobel** y **Roberts** son filtros diferenciales de primer orden y son considerados operadores dobles o de dos etapas. La detección de bordes se realiza en dos pasos, en el primero se aplica una máscara para buscar bordes horizontales, y en el segundo paso buscamos los verticales, el resultado final es la suma de ambos. Se muestran algunas máscaras de convolución comunes a continuación. Los detectores de fila (horizontales) son H_h y los detectores de columna (verticales) son H_v ([6]).

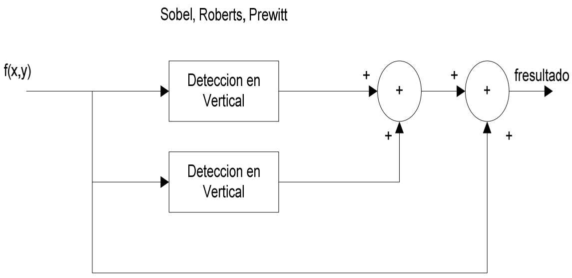


Figura 8. Filtros de Roberts, Prewitt y Sobel.

$$f_{Resultado}(x,y) = f(x,y) + ABS[f(x,y) * H_X(x,y)] + ABS[f(x,y) * H_Y(x,y)]$$

Tabla 5. Mascara del Filtro de Roberts

H_h	0	0	-1
	0	1	0
	0	0	0
H_v	-1	0	0
	0	1	0
	0	0	0

Tabla 6. Mascara del Filtro de Prewitt

H_h	1	0	-1
	1	0	-1
	1	0	-1
H_v	-1	-1	-1
	0	0	0
	1	1	1

Tabla 7. Mascara del Filtro de Sobel

H_h	1	0	-1
	2	1	-2
	1	0	-1
H_v	-1	-2	-1
	0	0	0
	1	2	1

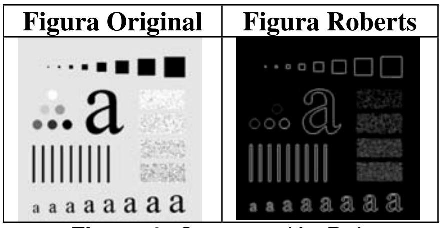


Figura 9. Comparación Roberts.

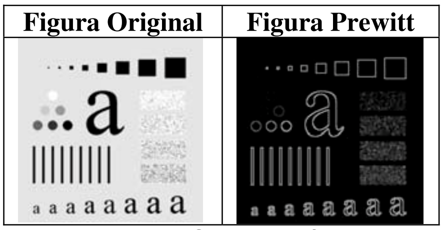


Figura 10. Comparación Prewitt.

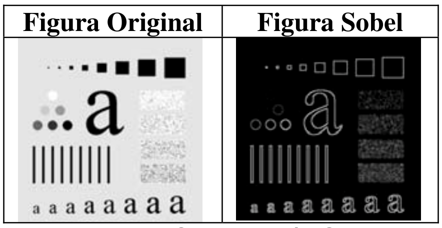


Figura 11. Comparación Sobel.

2.2.5. Enfatizador de Altas Frecuencias. El realce de la nitidez con este método consiste en sustraer una versión de la imagen original sin bordes, por ejemplo, el resultado de un filtro pasa bajos, de un arreglo de la misma imagen multiplicada por un factor escalar K, dicha sustracción debe resultar en la imagen original con los bordes ligeramente acentuados.

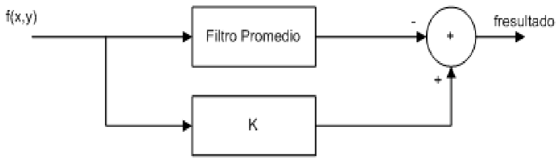


Figura 12. Enfatizador de Altas Frecuencias.

$$f_{Resultado}(x,y) = k[f(x,y)] - f(x,y) * H(x,y)$$

---

5 *http://www.des.udc.es/~adriana/TercerCiclo/CursoImagen/curso/web/Filtrado_Espacial_Borde.html*

6 *http://www.des.udc.es/~adriana/TercerCiclo/CursoImagen/curso/web/Filtrado_Espacial_Borde.html*

Tabla 8. Mascara del E. de Altas Frecuencias

0.1111	0.1111	0.1111
0.1111	0.1111	0.1111
0.1111	0.1111	0.1111

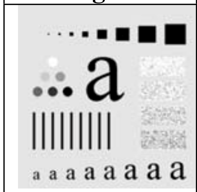
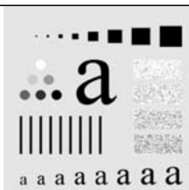
Figura Original	Figura E.A.F. K=2	Figura E. A.F. K=3
		

Figura 13. Comparación Enfatizador de Altas Frecuencias.

2.2.6. Laplaciano. El operador Laplaciano se define como un filtro diferencial de segundo orden, en general estos proporcionan una localización mejor del borde y los contornos detectados son curvas cerradas. Esto es muy importante en la segmentación de imagen. También, no hay respuesta a las áreas de variaciones lineales lisas en intensidad.

No obstante presenta una sensibilidad más grande frente al ruido, una ligera incapacidad para determinar la dirección de los bordes y puede generar bordes dobles.

El Laplaciano es omnidireccional, es decir, destacará los bordes en todas las direcciones, entonces producirá bordes más agudos que la mayoría de las otras técnicas.

El Laplaciano detecta los bordes, es decir, las altas frecuencias de la imagen, sin considerar la orientación, por lo que además de utilizarse para la detección de bordes sirve también para el filtrado paso alto de imágenes([7]).

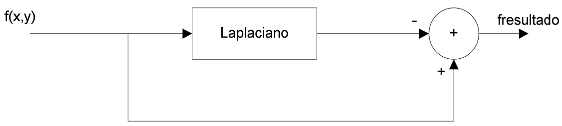


Figura 14. Filtro Laplaciano.

$$f_{\mathrm{Re}\,sultado}(x,y) = f(x,y) - f(x,y) * H(x,y)$$

Tabla 9. Mascara del Laplaciano

0.1667	0.6667	0.1667
0.6667	-3.3333	0.6667
0.1667	0.6667	0.1667

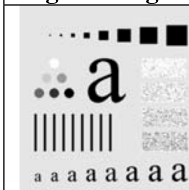
Figura Original	Figura Laplaciano
	

Figura 15. Comparación Laplaciano.

3. Ajuste de Curva

Es una ecuación dada en un grafico, dependiendo del grado de correlación que mas se ajuste al conjunto de datos.

3.1. Ajuste Logarítmico

Se busca ajustar la información de la imagen hacia una curva logarítmica, el cual se logra aplicando la función descrita en la figura 17. Donde k es una constante que da en valor de la curva

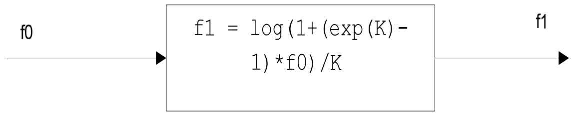


Figura 16. Ajuste Logarítmico.

Figura Original	Figura Ajuste Log K=2	Figura Ajuste Log K=3
		

Figura 17. Comparación Ajuste Logarítmico ([8]).

3.2 Ecualización de Histograma

La ecualización del histograma de una imagen es una transformación que pretende obtener para una imagen un histograma (*representación gráfica de una variable en forma de barras, donde el valor de cada barra es proporcional a la frecuencia de los valores representados*) con una distribución uniforme. Es decir, que exista el mismo número de pixeles para

[7] *http://www.des.udc.es/~adriana/TercerCiclo/CursoImagen/curso/web/Filtrado_Espacial_Borde.html*

[8] *FOTO: http://www3.uji.es/~lopeza/pdidoc//E75/figuras/figura-3-10.html*

cada nivel de gris del histograma de una imagen monocroma.

El resultado de la ecualización maximiza el contraste de una imagen sin perder información de tipo estructural, esto es, conservando su información ([9]).

$$v(i,j) = \frac{(Fu(u(i,j)) - Fu(a))}{1 - Fu(a)}(L-1) + 0.5$$

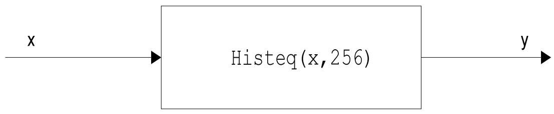


Figura 18. Ecualización de Histograma.

Donde HISTEG es una función especial de MATLAB que hace la ecualización de histograma.

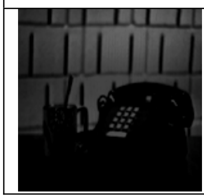
Figura Original	Figura E. de Histograma
	

Figura 19. Comparación Ecualización de Histograma.

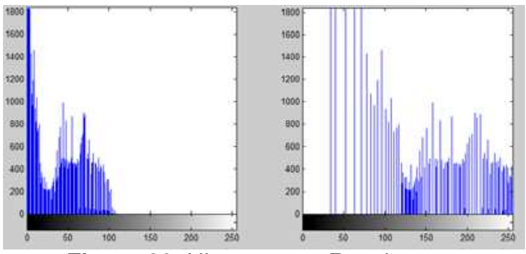


Figura 20. Histogramas Resultantes.

Se observa en el histograma de la derecha como los valores de la imagen se distribuyen mejor en la escala de grises, mejorando el contraste.

4. Implementación y resultados experimentales

Para el desarrollo del proyecto implementaremos los filtros y ajustes provistos en las secciones anteriores en una interfaz grafica de usuario, de esta manera obtendremos una aplicación, realizada en MATLAB, y la que puede convertirse en una herramienta de gran utilidad en el diagnostico de traumas en el cuerpo humano.

La aplicación mejora imágenes radiográficas y podría ayudar a detectar fracturas y fisuras que son imperceptibles a simple vista.

Al cargar una imagen radiográfica en el programa esta es llevada a escala de grises, debido a que el tratamiento de este tipo de imágenes es más simple, y además de que se puede apreciar de mejor manera los perfiles de los huesos, que son el principal objetivo de nuestra propuesta.

A continuación se presentan algunos resultados obtenidos al trabajar con la aplicación:

Original	Filtrada 1era UnSharp
	
Filtrada 2da	**Filtrada 3era UnSharp**
	

Figura 21. Ejemplo 1

Original	Ajuste Logarítmico K = 2.08	Enfatizador de Altas Frecuencias A = 2.3
		

Figura 22. Ejemplo 2

[9] http://es.wikipedia.org/wiki/Ecualizaci%C3%B3n_del_histograma

Original	Filtrada Por Sobel
	

Figura 23. Ejemplo 3

Original	Original con Zoom	Prewitt y Enfatizador de Altas Frecuencias con A = 2
		

Figura 24. Ejemplo 4

Según los resultados obtenidos en las figuras 21, 22, 23 y 24, se muestra que de ser necesario se podrían re-utilizar los resultados previos obtenidos, con el fin de alcanzar los resultados deseados, o bien se podría aceptar el primer filtrado, según lo requiera el usuario.

5. Conclusiones

El programa desarrollado compila numerosas herramientas aprendidas en el curso dando una aplicación didáctica como la mejora de imágenes radiográficas.

El tratamiento de las imágenes con estas herramientas ayudaran o confirmar o descartar cualquier duda que el doctor pudiera tener acera sobre una determinada fisura o fractura, cabe destacar que los problemas comunes radican en meseta tibial de la rodilla, el escafoides de la muñeca y fractura imperceptibles en el cuello del fémur especialmente en ancianos.

El usuario debe familiarizarse con el programa ya que la aplicación de un determinado filtro no necesariamente sería suficiente para encontrar una mejora significativa de la imagen, todo depende de la imagen fuente

Con la utilización de este software para imágenes de alta resoluciones (ejemplo: 3264 x 2448), Matlab comienza a inhibirse y deja de tener un funcionamiento esperado, los cálculos para esa cantidad de pixeles exigen de una cantidad grande de memoria

Los filtros utilizados son de tipo pasa altos, debido a que las altas frecuencias de una imagen corresponden a los filos bien definidos de las mismas, de hecho en el dominio del espacio es equivalente al rápido cambio del brillo de 2 o más pixeles adyacentes.

Para imágenes de pobre resolución (256x256), el filtrado por cualquiera de los tipos mostrados aquí, aumentan el ruido de la misma, provocando distorsión y carece de todo propósito el funcionamiento de este programa para ese tipo de imágenes.

Los ajustes de curva ya sea exponencial o logarítmica (como en este caso) llevan los pixeles a una función, la cual determinara su nueva intensidad según sea la grafica a la que se esté evaluando. El ajuste logarítmico aplicado es ideal para radiografías opacas o con poca luz porque ayuda a obtener mejores detalles ocultos, debido a los problemas de ojo humano al solo poder diferencia 80 escala de grises, mientras más alto sea el valor de k más brillo tendrá la imagen.

El programa efectivamente ayudo al traumatólogo Dr. Miguel Mite a confirmar o descartar fracturas y fisuras presentes en sus pacientes con mayor precisión y seguridad.

6. Referencias

[1] Fotografías. Dr. Miguel Mite Vivar, Medico Traumatólogo, Clínica San Francisco, Ciudad de Guayaquil.

[2] Erik Valdemar Cuevas Jimenez ,"Visión por Computador utilizando MatLAB y el Toolbox de Procesamiento Digital de Imágenes", 2007

[3] Hector Zarate, "Aplicación de las Matrices en el Procesamiento de Imágenes", Universidad Nacional Autónoma de México

[4] Alexander Vera Tasamá, Andrés F. Ramírez Sánchez, Daniel Moreno Martínez, Milton J. Alzate Silva, "Aplicación Didáctica para el Procesamiento de Imágenes Digitales Usando Interfaz Gráfica de Usuario en MATLAB",

[5] Departamento de Ingeniería electrónica, Telecomunicación y Automática. "Detección de bordes en una imagen", Universidad de Jaen, 2005

[6] Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins, "Digital Image Processing using Matlab", Prentice Hall, 2004

[7] Vinay K. Ingle, John G. Proakis, Northeastern University, "Digital Signal Processing using Matlab V.4", PWS, 1997

[8] Steven W. Smith, "A Practical Guide for Enginners and Scientist" , Newnes 2003