

Universidad de La Habana  
Facultad de Matemática y Computación



# Detección de patrones en imágenes de mamografías digitales

Autor:

**Miguel Alejandro Asin Barthelemy**

Tutores:

**MSc. Damian Valdés Santiago**

Trabajo de Diploma  
presentado en opción al título de  
Licenciado en (Matemática o Ciencia de la Computación)

28 de Diciembre de 2022

[github.com/M4S1N/thesis](https://github.com/M4S1N/thesis)

*Dedico este logro a la persona que me ha dado la fuerza para seguir adelante pese a  
todos los tropiezos.*

# Agradecimientos

A mi familia por apoyarme en todo momento.

A mis compañeros por los buenos y malos momentos que pasamos juntos.

A mi tutor Damián por creer en que lo podía lograr.

# Opinión del tutor

Opiniones de los tutores

# Resumen

Se propone el desarrollo de un algoritmo para determinar una base wavelet que permita la detección de anomalías en mamografías digitales, para ello definen un conjunto de conceptos pertenecientes a la teoría wavelet que serán usados para introducir un nuevo tipo llamado Shapelet. Las Shapelet son wavelet que a diferencia de otras wavelets tienen en cuenta la forma de la señal que está analizando. Se muestra un algoritmo para crear una base Shapelet a partir de un patrón y usada para detectar dicho patrón en una señal de muestra, para ello se construye un sistema de ecuaciones no lineal cuya solución la constituye el vector que se usará de base. Con estos conceptos y una introducción a las wavelets en dos dimensiones se propone una forma de extender las Shapelet al análisis de imágenes. Posteriormente se adapta el sistema de ecuaciones a dos dimensiones teniendo en cuenta las traslaciones diádicas que se producen durante el cálculo de la transformada y, tras resolver el sistema se obtiene nuevamente la base Shapelet. Como experimento se toma una colección de mamografías de las cuales algunas presentan algún tipo de anomalía y se le aplica el algoritmo propuesto. Evaluando de esta forma el modelo creado a partir de los resultados obtenidos.

# Abstract

The development of an algorithm to determine a wavelet base that allows the detection of anomalies in digital mammograms is proposed, for this they define a set of concepts belonging to the wavelet theory that will be used to introduce a new type called Shapelet. Shapelets are wavelets that, unlike other wavelets, take into account the shape of the signal being analyzed. An algorithm is shown to create a Shapelet base from a pattern and used to detect said pattern in a sample signal, for this a non-linear system of equations is built whose solution is the vector that will be used as the base. With these concepts and an introduction to wavelets in two dimensions, a way to extend Shapelets to image analysis is proposed. Subsequently, the system of equations is adapted to two dimensions taking into account the dyadic translations that occur during the calculation of the transform and, after solving the system, the Shapelet basis is obtained again. As an experiment, a collection of mammograms is taken, some of which present some type of anomaly, and the proposed algorithm is applied. Evaluating in this way the model created from the results obtained.

# Índice general

|   |           |
|---|-----------|
| <b>Introducción</b>   | <b>1</b>  |
| <b>1. Estado del Arte</b>                                     | <b>3</b>  |
| 1.1. Introducción a las Wavelets . . . . .                    | 4         |
| 1.1.1. Definiciones . . . . .                                 | 5         |
| 1.1.2. Banco de Filtros . . . . .                             | 7         |
| 1.1.3. Trasformada Discreta de Shapelet . . . . .             | 9         |
| 1.2. Wavelets en dos dimensiones . . . . .                    | 14        |
| 1.2.1. Definiciones . . . . .                                 | 14        |
| 1.2.2. Banco de filtros . . . . .                             | 15        |
| <b>2. Propuesta</b>   | <b>18</b> |
| 2.1. Definición del filtro . . . . .                          | 18        |
| 2.2. Cálculo del filtro . . . . .                             | 20        |
| <b>3. Detalles de Implementación y Experimentos</b>           | <b>23</b> |
| 3.1. Métodos para resolver sistemas de ecuaciones . . . . .   | 24        |
| 3.1.1. Método de Newton . . . . .                             | 24        |
| 3.1.2. Método de Cuasi Newton . . . . .                       | 25        |
| 3.1.3. Método de Descenso por Gradiente . . . . .             | 25        |
| 3.1.4. Método de Runge Kutta . . . . .                        | 26        |
| 3.1.5. Gekko . . . . .  | 27        |
| 3.1.6. Mejores Resultados . . . . .                           | 28        |
| 3.2. Algoritmo de reconocimiento en una dimensión . . . . .   | 29        |
| 3.3. Algoritmo de reconocimiento en dos dimensiones . . . . . | 31        |
| <b>Conclusiones</b>   | <b>38</b> |
| <b>Recomendaciones</b>  | <b>39</b> |
| <b>Bibliografía</b>   | <b>40</b> |

# Índice de figuras

|  |    |
|--|----|
| 1.1. Izquierda: Onda(Fourier). Derecha: Ondita(Wavelet) . . . . .  | 5  |
| 1.2. Coeficientes ortogonales de Daubechies . . . . .  | 11 |
| 1.3. Ejemplo de una señal patrón ( $m$ ). . . . .  | 12 |
| 1.4. Señal de muestra con el patron $m$ incluido. . . . .  | 13 |
| 1.5. Representación de los vectores de aproximación (los primeros 32 valores) y detalle (los valores entre 32 y 63) y, la medida de similaridad con el patrón. . . . .                                     | 14 |
| 3.1. Efectividad (%) del método de Newton en el modelo para un aumento de la dimensión. . . . .  | 24 |
| 3.2. Efectividad (%) del método de Cuasi Newton en el modelo para un aumento de la dimensión. . . . .  | 25 |
| 3.3. Efectividad (%) del método de Descenso por Gradiente en el modelo para un aumento de la dimensión. . . . .  | 26 |
| 3.4. Efectividad (%) del método de RungeKutta en el modelo para un aumento de la dimensión. . . . .  | 27 |
| 3.5. Efectividad (%) del método de Gekko en el modelo para un aumento de la dimensión. . . . .   | 27 |
| 3.6. Resultados obtenidos empleando los metodos del Descenso por Gradiente y Runge Kutta para encontrar una aproximación inicial y, los métodos Newton, Cuasi Newton y Gekko para resolver el sistema. . . | 28 |
| 3.7. Vector $cD$ obtenido tras aplicar la DST (naranja) y, medida de similaridad con $\alpha = 0,1$ (azul). . . . .  | 30 |
| 3.8. Precisión y Recobrado del algoritmo de una dimensión para distintos valores del umbral usando la mediada $\mathbb{S}$ . . . . .   | 31 |
| 3.9. Ejemplo de un patrón y una imagen de prueba. . . . .  | 34 |
| 3.10. Imagenes de aproximación, detalles horizontales, detalles verticales, detalles (comúnmente llamados diagonales), imagen con el patrón señalado y medida $\mathbb{S}$ , en ese orden. . . . .         | 35 |
| 3.11. Resultados obtenidos en el proceso de detección de la anomalía en la mamografía. . . . .   | 36 |



|  |    |
|--|----|
| 3.12. Valores de precisión y recobrado obtenidos durante el aumento de los<br>datos procesados, de una colección de 200 mamografías. . . . . | 37 |
|--|----|

# Ejemplos de código

|   |    |
|---|----|
| 3.1. Creación de una wavelet a partir del banco de filtros . . . . .      | 29 |
| 3.2. Cálculo de la transformada discreta de Wavelet usando pywt . . . . . | 29 |
| 3.3. Leer una imagen .dcm . . . . .                                       | 32 |

# Introducción

Cuando nos referimos a las principales causas de muerte por cáncer en el mundo, el cáncer de mama ocupa uno de los primeros lugares entre las mujeres. Según datos proporcionados por la Organización Mundial de la salud, en el 2020 el cáncer de mama ocupó el primer lugar en cantidad de casos con 2,26 millones y el quinto en defunciones con 685 mil muertes[3]. En el caso de América, el cáncer de mama constituye la principal causa de muerte por cáncer, dadas cifras también del 2020, se diagnosticaron 210 mil nuevos casos y casi 68 mil defunciones.

Según los especialistas el cáncer de mama se presenta con mayor frecuencia como una masa indolora en la mama, el cuál puede desarrollarse por razones distintas al cáncer. Otros síntomas del cáncer de mama incluyen engrosamiento de la mama, alteración en el tamaño, la forma o la apariencia de la mama, alteraciones de la piel como enrojecimiento, picaduras o hoyuelos, cambio en la apariencia del pezón o la piel alrededor (areola), y/o secreción anormal del pezón[4].

Sin embargo el tratamiento del cáncer de mama puede ser efectivo en caso de detectarse a tiempo, mediante el uso de mamografías de detección. Estas mamografías son analizadas por un radiólogo quien se encarga de dar una clasificación (de 0 a 6) basado en un sistema estándar llamado Breast Imaging Reporting and Data System o BI-RADS. En varios casos es posible calificar una radiografía con categoría cero, esto se debe a que el radiólogo pudo haber visto una posible anomalía, pero que no está definida con claridad y que podrían necesitarse exámenes adicionales. En dependencia de los recursos de los cuales se dispone este hecho puede representar un problema, pues no podría ser posible la creación de nuevas mamografías y de caso de ser positiva la muestra al cáncer de mama, podría desembocar en un agravamiento de la condición.

Teniendo en cuenta este problema, se han buscado formas de obtener una calificación más acertada mediante el uso de equipos de cómputo. La cual podría constituir una vía factible teniendo en cuenta el desarrollo actual en el campo del procesamiento de imágenes digitales y la detección de patrones.

En este sentido se han creado un gran número de algoritmos para el reconocimiento de patrones, con todo tipo de aplicaciones en la sociedad actual, las cuales van desde detección de rostros en fotografías hasta el análisis y reconocimiento de estrellas y galaxias. Haciendo usos de técnicas de inteligencia artificial y suficientes datos de

---

muestra, ha sido posible entrenar varios modelos de clasificación que permiten la detección efectiva de las características para las cuales fue desarrollado.

Un segundo método consiste en el uso de wavelets para la detección de patrones. Las wavelets son un conjunto de funciones con la peculiaridad de que son muy efectivas a la hora de analizar una señal en tiempo y frecuencia. Si se tiene en cuenta que una imagen puede ser interpretada como una señal bidimensional, es posible ampliar el concepto de las wavelets al análisis de imágenes o lo que es en este caso, mamografías digitales.

Se desea entonces crear una base de funciones wavelets que permita analizar un patrón de una anomalía o tumor cancerígeno y que sean capaces de detectarlo en mamografías digitales de muestra. Para ello será necesario entender el concepto de wavelet, como podrían emplearse en la detección de patrones unidimensionales y, posteriormente extender estos conceptos a dos dimensiones para tratar con imágenes o mamografías.

A continuación se brindará un panorama en el cual se gestó la idea del uso de las funciones wavelet para el reconocimiento de patrones. Se propondrá un tipo de wavelet que se definen específicamente para el trato con patrones; característica que resultará muy útil en la tarea de reconocimiento en imágenes. En una segunda instancia, se analizarán un conjunto de métodos utilizados para la resolución de sistemas de ecuaciones con el objetivo de encontrar el que mejor compatibilidad tenga con el modelo que se usará y, posteriormente, se emplearán las herramientas adquiridas para desarrollar una extensión hacia dos dimensiones del tipo de wavelet definida en una dimensión y un algoritmo que permita la detección del patrón.

Por último se presentará un pequeño programa que lleve a la práctica los conceptos que se analizarán tanto para una como para dos dimensiones, en correspondencia con los resultados obtenidos por el programa para distintos casos que podrían presentarse durante la tarea de reconocimiento. Se dará una evaluación del método propuesto y como podría mejorarse para alcanzar una mejor precisión, resultado que podría culminar con la creación de una gran herramienta que pueda ser utilizado por radiólogos para la detección temprana de esta peligrosa enfermedad.

# Capítulo 1

## Estado del Arte

Durante la década de los 70 un grupo relativamente pequeño de investigadores se involucró en el tema tratando al análisis de imágenes médicas como un problema de procesamiento de información en una única imagen. Esta línea posee enfoques basados en reconocimiento de patrones, procesamiento de imágenes y/o señales y visión computarizada. En 1973 Sklansky y Ballard[1] describen un método por el cual una computadora realiza los bordes de las imágenes de tumores en radiografías y escaneos de isótopos. El algoritmo extendido detecta el mejor conjunto de curvas cerradas localmente convexas que contienen candidatos a tumores. Por lo tanto, se desarrolló una técnica de selección con criterios subjetivos para clasificar a los candidatos y seleccionar los que probablemente fueran tumores. Según los investigadores, aunque la naturaleza de la imagen de datos dificultó la mejora de los bordes debido a la tecnología de esta época, se obtuvieron resultados positivos.

Otros esfuerzos, como el trabajo de Pizer y Todd-Pokropek[14] en 1978, enfatizaron el mejoramiento de la imagen y las estrategias de visualización usando técnicas lineales y estacionarias para corregir el ruido y la degradación, observando que estos eran problemas críticos para los usuarios finales (radiólogos y otros). Si bien no clasifican ni automatizan nada sentó las bases del preprocesamiento.

Hacia la década de 1980 se realizaron varias investigaciones cuya característica fundamental fue el desarrollo de ideas que se basaban en la detección de bordes por contrastes en bancos de datos bidimensionales y luego la aplicación de un agrupamiento o unión básica de bordes utilizando algún tipo de heurística de búsqueda de contorno basándose en las propiedades de suavidad embebidas en la figura a analizar, tal y como se expone en el trabajo de Yachida[12] en 1980. Estos enfoques aprovecharon algunos avances generales hechos por la comunidad científica orientada al procesamiento de imágenes y visión computarizada, y podría ser visto como el precursor de la variedad de enfoques de búsqueda de bordes deformables presentes en el desarrollo de hoy en día.

Los primeros intentos de sistemas de diseño asistido por computador (Computer Aided Design, CAD por sus siglas en inglés) completamente automático en mamografías de rayos X fueron propuestos recién sobre 1987 (Chan et al.[2], 1987). En el mismo se investigó la aplicación de algoritmos para la detección de microcalcificaciones en mamografías digitales. El sistema de detección de la anomalía se basó en una técnica de diferenciación entre una imagen con señal suprimida y una imagen con señal mejorada para eliminar el fondo estructurado en la mamografía. Para ello se empleó el método de Monte Carlo, con el fin de generar grupos de microcalcificaciones que se superponen en el fondo de la mamografía. Esto permitió una evaluación de la precisión de detección del método y la dependencia de esta precisión de las características físicas de las microcalcificaciones.

Como parte del proceso de mejoramiento de la imagen, se empleó un filtro espacial que estuviese correlacionado con el tamaño y características de la microcalcificación. Aunque no es un tema que se expuso directamente en la investigación, se comenzó a desarrollar la idea del uso de wavelets para la detección de anomalías, lo que dio paso a que en los inicios del siglo actual se comenzaran a realizar nuevos proyectos de procesamiento digital empleando las funciones Wavelets.

En 2005 José Salvado y Bruno Roque[10] publican un artículo sobre el uso del análisis wavelet para la detección de calcificaciones en mamografías digitales. Se aplica el concepto de filtro, descomposición y reconstrucción de la imagen característico de las wavelets a fin de eliminar el ruido en la mamografía digital y obtener una muestra mejorada que permita una detección efectiva de la anomalía.

Muchos son los algoritmos que se han desarrollado para el procesamiento de imágenes y detección de patrones que hacen uso de técnicas de *machine learning*, sin embargo el empleo de wavelets para la detección de patrones es una técnica que no se ha explotado lo suficiente en comparación al método anterior, a lo que salta la pregunta, ¿qué son las wavelets y cómo podrían usarse en la detección de patrones?

## 1.1. Introducción a las Wavelets

El origen de la teoría wavelets se remonta a los análisis armónicos del campo de las matemáticas puras desarrollados por el matemático francés Jean Baptiste Joseph Fourier (1768-1830), conocido por ser el primero en crear un método para expresar toda función periódica como suma de funciones trigonométricas (senos y cosenos) llamada Series Trigonómicas de Fourier.

A inicios del siglo XX, Alfred Haar desarrolló un conjunto de funciones que más adelante se nombrarían en su honor: Wavelets Haar. Estas constituyen el caso más simple conocido de funciones wavelets y pueden ser utilizadas para el análisis de una señal en tiempo y frecuencia debido a la propiedad que tienen de tener una mayor localización que las funciones armónicas utilizadas en el análisis de Fourier.

Más tarde, Paul Levy (1886-1971) demostró que las Wavelets de Haar, debido a su propiedad de escalado, brindan una mejor forma de modelar las funciones que el análisis de Fourier.[7]

En 1984, Jean Morlet en su intento de ayudar a geólogos para encontrar herramientas más efectivas que permitieran la localización de petróleo, desarrolló una forma de analizar las señales sísmicas para crear componentes que estuviesen bien localizadas en el espacio, llamándolas onditas, que más tarde serían conocidas como las wavelets de Morlet.

Hacia 1986 y 1987 la demostración de que la teoría wavelets aparece implícita en el análisis multirresolución por parte de Stéphane Mallat en 1986 y el descubrimiento en 1987 de Ingrid Daubechies de una nueva clase de wavelet que no solo era ortogonal si no que podía ser implementada usando ideas de filtrado digital simple, provocaron el detonante en la revolución wavelet.[13]

### 1.1.1. Definiciones

La palabra Wavelet significa onda pequeña u ondita. Matemáticamente quiere decir que la onda es de corta duración, esta característica se conoce como localización y es una de las ventajas que proporcionan las funciones wavelets sobre la serie de Fourier, ver figura 1.1.

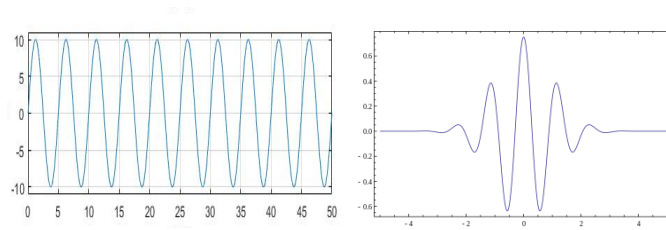


Figura 1.1: Izquierda: Onda(Fourier). Derecha: Ondita(Wavelet)

**Definición 1.1** De manera formal, una wavelet es una función  $\Psi(t) \in \ell^2(\mathbb{R})$  tal que la familia de funciones:

$$\Psi_{a,b}(t) := \frac{1}{\sqrt{a}} \Psi\left(\frac{t-b}{a}\right)$$

donde  $a$  y  $b$  son reales que indican la escala y el desplazamiento respectivamente, es una base ortonormal en el espacio de Hilbert  $\ell^2(\mathbb{R})$ .

Este conjunto de funciones están definidas en todo  $\mathbb{R}$ , por lo que representan funciones “continuas”. Aunque esta forma de expresar las definición de funciones wavelets

es muy útil para el análisis, fundamentalmente teórico, de señales en investigaciones científicas, puede presentar un problema para la codificación de las mismas en campos como la ingeniería y las ciencias computacionales. Por ello se brinda una definición en  $\mathbb{Z}_N$  un poco más tratable para equipos de cómputo y será la que se utilizará en lo adelante.

**Definición 1.2** *Se define la función wavelet discreta  $\Psi(t) \in \ell^2(\mathbb{Z}_N)$  tal que la familia de funciones:*

$$\Psi_{j,k}(t) := 2^{\frac{j}{2}} \Psi(2^j t - k)$$

*con  $j$  y  $k$  enteros, es una base ortonormal en el espacio  $\ell^2(\mathbb{Z}_N)$ .*

Como se puede apreciar la definición de wavelet discreta se obtiene al discretizar los parámetros de desplazamiento y escalamiento dentro de la definición de wavelet continua:

$$\begin{aligned} a &= 2^{-j} \\ b &= k2^{-j} \quad \text{con } j, k \in \mathbb{Z} \end{aligned}$$

A la función  $\Psi$  se le llama entonces wavelet madre. En su forma discreta cada elemento de la base wavelet constituye un vector de dimensión  $N$ . Veamos entonces como es posible obtener esta base y por tanto wavelet madre. A continuación se presentan un conjunto de definiciones a fin de proporcionar una notación y un algoritmo para obtener dichas wavelets.

**Definición 1.3** *Se define la reflexión conjugada  $\tilde{w}$  de un vector  $w \in \ell^2\mathbb{Z}_N$  como:*

$$\tilde{w}(n) = \overline{w(-n)} = \overline{w(N-n)}$$

**Definición 1.4** *Se define la convolución  $*$  entre dos vectores  $w, z \in \ell^2(\mathbb{Z}_N)$  como:*

$$(z * w)(n) = \sum_{i=0}^{N-1} w(i)z(n-i)$$

**Definición 1.5** *Sea  $w \in \ell^2(\mathbb{Z}_N)$  un vector y  $k \in \mathbb{Z}$ , se define  $R_k w$  como la traslación de  $w$  por  $k$  dado por:*

$$R_k w(n) = w(n-k)$$



### 1.1.2. Banco de Filtros

**Definición 1.6** Sea  $N$  divisible por  $2^p$ . Se define un banco de filtros wavelet de  $p$  escalas según dos sucesiones de vectores  $u_1, u_2, \dots, u_p$  y  $v_1, v_2, \dots, v_p$ , tales que:

$$u_j, v_j \in \ell^2\left(\mathbb{Z}_{\frac{N}{2^{p+j-1}}}\right), \quad \forall j = 1, 2, \dots, p,$$

y tales que la matriz del sistema:

$$A_j(m) = \frac{1}{\sqrt{2}} \begin{bmatrix} \hat{u}_j(m) & \hat{v}_j(m) \\ \hat{u}_j\left(m + \frac{N}{2^j}\right) & \hat{v}_j\left(m + \frac{N}{2^j}\right) \end{bmatrix}$$

es unitaria  $\forall m = 0, 1, \dots, \frac{N}{2^j} - 1, j = 1, 2, \dots, p$ .

Considerese esta última definición. Sea  $u_j \in \ell^2(\mathbb{Z}_{\frac{N}{2^{j-1}}})$  un vector tal que  $\{R_{2^k}u_j\}_{k=0}^{\frac{N}{2^j}-1}$  constituye una base ortonormal. Si se define  $v_j \in \ell^2(\mathbb{Z}_{\frac{N}{2^{j-1}}})$  tal que:

$$v_j(n) = (-1)^{n-1} \overline{u_j(1-n)}$$

entonces se cumple que  $\{R_{2^k}u_j\}_{k=0}^{\frac{N}{2^j}-1} \cup \{R_{2^k}v_j\}_{k=0}^{\frac{N}{2^j}-1}$  es una base wavelet en  $\ell^2(\mathbb{Z}_{\frac{N}{2^{j-1}}})$  y por tanto la matriz del sistema  $A_j(m)$  es unitaria para toda  $m = 0, 1, \dots, \frac{N}{2^j} - 1$  (ver demostración en [5]).

**Definición 1.7** Suponga que  $N$  es divisible por  $2^p$ . Se define  $D^p : \ell^2(\mathbb{Z}_N) \rightarrow \ell^2(\mathbb{Z}_{\frac{N}{2^p}})$  para  $z \in \ell^2(\mathbb{Z}_N)$  como:

$$D^p(z)(n) = z(2^p n), \quad \text{para } n = 0, 1, \dots, M-1$$

Al operador  $D$  se le denomina operador de submuestreo.

**Definición 1.8** Suponga que  $N$  es divisible por  $2^p$ . Se define  $U^p : \ell^2(\mathbb{Z}_{\frac{N}{2^p}}) \rightarrow \ell^2(\mathbb{Z}_N)$  para  $z \in \ell^2(\mathbb{Z}_{\frac{N}{2^p}})$  como:

$$U^p(z)(n) = \begin{cases} z\left(\frac{n}{2^p}\right) & \text{si } n \text{ es divisible por } 2^p \\ 0 & \text{en otro caso} \end{cases}$$

Al operador  $U$  se le denomina operador de sobremuestreo.

**Definición 1.9** Suponga que  $N$  es divisible por  $2^p$ . Sean  $u_j, v_j \in \ell^2(\mathbb{Z}_{\frac{N}{2^{j-1}}})$ . Se definen  $f_\ell, g_\ell \in \ell^2(\mathbb{Z}_N)$  como:

$$\begin{aligned} f_\ell &= g_{\ell-1} * U^{\ell-1}(v_\ell) \\ g_\ell &= g_{\ell-1} * U^{\ell-1}(u_\ell) \end{aligned}$$

para toda  $\ell = 2, 3, \dots, p$ , donde  $f_1 = v_1$  y  $g_1 = u_1$ .

Considere ahora que los vectores  $u_\ell$  y  $v_\ell$  de la definición anterior son tomados de tal forma que la matriz del sistema  $A_\ell(n)$  es unitaria para toda  $n = 0, 1, \dots, \frac{N}{2^\ell} - 1$ . Entonces se cumple que:

$$\{R_{2^\ell k} f_\ell\}_{k=0}^{\frac{N}{2^\ell}-1} \cup \{R_{2^\ell k} g_\ell\}_{k=0}^{\frac{N}{2^\ell}-1}$$

constituye una base ortonormal en con  $\frac{N}{2^{\ell-1}}$  elementos (ver demostración en [6]).

Como consecuencia del resultado anterior si se desarrolla el término  $g_{\ell-1}$  en cada nueva ecuación se obtiene la siguiente base de  $\ell^2(\mathbb{Z}_N)$ :

$$\{R_{2^k} f_1\}_{k=0}^{\frac{N}{2}-1} \cup \{R_{4^k} f_2\}_{k=0}^{\frac{N}{4}-1} \cup \dots \cup \{R_{2^p k} f_p\}_{k=0}^{\frac{N}{2^p}-1} \cup \{R_{2^p k} g_p\}_{k=0}^{\frac{N}{2^p}-1}$$

A partir de la definición 1.2 se cumple que, si se toma  $\Psi_{j,k} = R_{2^j k} f_j$  y  $\Phi_{j,k} = R_{2^j k} g_j$  entonces  $f_1$  y  $g_1$  constituyen la wavelet madre y la wavelet padre respectivamente.

Durante el proceso de análisis de una señal se aplica la llamada Transformada Discreta de Wavelet esta consiste en determinar los coeficientes wavelets de la señal respecto a la base representada por el banco de filtros. Estos coeficientes coinciden con los valores de  $z * \tilde{v}$  y  $z * \tilde{u}$ , más aún, se cumple que:

$$v * U(D(z * \tilde{v})) + u * U(D(z * \tilde{u})) = z$$

representa todo el proceso de análisis y síntesis de una señal  $z$  usando el banco de filtros obtenido a partir de  $u$  y  $v$ .

Como se ha podido apreciar hasta el momento, encontrar una wavelet en  $\ell^2(\mathbb{Z}_N)$  se reduce a encontrar un banco de filtros en  $\ell^2(\mathbb{Z}_N)$ . Así por ejemplo si se toman:

$$\begin{aligned} u &= \left( \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0, \dots, 0 \right) \\ v &= \left( \frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, 0, \dots, 0 \right) \end{aligned}$$

se cumple que  $\{R_{2k}v\}_{k=0}^{\frac{N}{2}-1} \cup \{R_{2k}u\}_{k=0}^{\frac{N}{2}-1}$  constituye una base ortonormal en  $\ell^2(\mathbb{Z}_N)$  y por tanto la matriz del sistema es unitaria, luego es posible afirmar que se obtiene un banco de filtros a partir de  $u$  y  $v$ . La wavelet que se obtiene a partir de este, es conocida como Wavelet Haar.

Hasta el momento se ha explicado que son las wavelets, se conocen las ventajas que posee en el análisis de señales respecto a las series de Fourier y que cada wavelet se reduce a encontrar un banco de filtros. Centrémonos entonces en este último aspecto; la diferencia que existe entre las wavelets está dada por los valores de los vectores  $u$  y  $v$  que son denominados generalmente como filtro de reconstrucción pasa bajo y filtro de reconstrucción pasa alto respectivamente. Los valores de estos filtros son calculados en dependencia de la finalidad para la cual se haya creado la wavelet, esto quiere decir que si se deseara crear una wavelet capaz de reconocer un patrón específico en una señal sería necesario crear un banco de filtros basado en los valores de dicho patrón. Para ello existen unas wavelets conocidas con el nombre de Shapelets.

### 1.1.3. Transformada Discreta de Shapelet

En 2017 Rodrigo Capobianco Guido[8] introduce la segunda generación de la Transformada Discreta de Shapelet (Discrete Shapelet Transform, DST-II por sus siglas en inglés). Esta wavelet tiene la característica de que a diferencia de otras wavelets conocidas, fusiona tres tipos de información: tiempo, frecuencia y forma. Se propone un procedimiento mediante el cual es posible obtener los vectores  $u$  y  $v$  del banco de filtros con un enfoque más simple al de su predecesor, la DST-I o simplemente DST[9].

El algoritmo propuesto para determinar el banco de filtros de la Shapelet tiene gran similitud al usado para determinar los filtros de la wavelet Daubechies. Por ello se muestra en una primera instancia el algoritmo para encontrarlos en caso de esta última.

El cálculo de un banco de filtros puede ser reducido a la determinación de un único vector  $v$  conocido como filtro pasa alto. A partir del cual, de cumplir con las condiciones consideradas en la definición 1.6, es posible determinar el valor del filtro pasa bajo  $u$  y por tanto el banco de filtros.

La primera condición que debe cumplir el vector  $v$  en la wavelet Daubechies es la condición de energía unitaria, es decir:

$$\sum_{k=0}^{N-1} v(k)^2 = 1 \quad (1.1)$$

La ecuación anterior esta muy relacionada a la segunda condición que es la condición de ortogonalidad, pues constituye a la vez la primera ecuación de este conjunto de igualdades. Se dan entonces  $\frac{N}{2} - 1$  ecuaciones relacionadas a las condiciones de ortogonalidad del filtro:

$$\sum_{k=0}^{N-2l-1} v(k)v(k+2l) = 0, \quad \forall l = 1, 2, \dots, \frac{N}{2} - 1 \quad (1.2)$$

Como tercera y última condición, se le imponen  $\frac{N}{2}$  ecuaciones de momento de desvanecimiento, también llamados momentos nulos:

$$\sum_{k=0}^{N-1} v(k)k^b = 0, \quad \forall b = 0, 1, \dots, \frac{N}{2} - 1 \quad (1.3)$$

Como se puede apreciar la suma total de las ecuaciones es  $N$ , donde  $N$  debe ser un número par, que coincide con el número de variables (componentes del vector  $v$ ). Por lo tanto se obtiene un sistema al menos compatible determinado y, en el peor caso, compatible con un número infinito de vectores posibles como candidatos al banco de filtro.

Una vez obtenido el valor de  $v$  aplicando algunos de los métodos de resolución de sistemas de ecuaciones conocidos, se determina el vector  $u$  a partir de la consideración dada en la definición 1.6. Se sabe que es posible tomar  $v$  a partir de  $u$  para que  $\{R_{2k}u\}_{k=0}^{\frac{N}{2}-1} \cup \{R_{2k}v\}_{k=0}^{\frac{N}{2}-1}$  constituya una base ortonormal de  $\ell^2(\mathbb{Z}_N)$  de la siguiente forma:

$$v(n) = (-1)^{n-1} \overline{u(1-n)},$$

sea entonces  $m = 1 - n$ , se tiene:

$$\begin{aligned} v(1-m) &= (-1)^{-m} \overline{u(m)} \\ (-1)^m v(1-m) &= \overline{u(m)} \\ (-1)^m \overline{v(1-m)} &= u(m) \end{aligned}$$

A partir de los vectores  $u$  y  $v$ , se tiene el banco de filtros y a su vez la wavelet correspondiente. La figura 1.2 muestra una tabla con los posibles valores de  $u$  en el caso de la wavelet Daubechies para diferentes valores de  $N$ .

Como se puede apreciar las Daubechies obtenidas serán las mismas para un valor de  $N$  dado y no se modificarán independientemente de la señal que se esté analizando. Si se desea realizar la tarea de detectar un patrón en una señal no sería factible utilizar

| D2 (Haar) | D4         | D6          | D8          | D10                         | D12                          | D14                          | D16                          | D18                          | D20                          |
|-----------|------------|-------------|-------------|-----------------------------|------------------------------|------------------------------|------------------------------|------------------------------|------------------------------|
| 1         | 0.6830127  | 0.47046721  | 0.32580343  | 0.22641898                  | 0.15774243                   | 0.11009943                   | 0.07695562                   | 0.05385035                   | 0.03771716                   |
| 1         | 1.1830127  | 1.14111692  | 1.01094572  | 0.85394354                  | 0.69950381                   | 0.56079128                   | 0.44246725                   | 0.34483430                   | 0.26612218                   |
|           | 0.3169873  | 0.650365    | 0.89220014  | 1.02432694                  | 1.06226376                   | 1.03114849                   | 0.95548615                   | 0.85534906                   | 0.74557507                   |
|           | -0.1830127 | -0.19093442 | -0.03957503 | 0.19576696                  | 0.44583132                   | 0.66437248                   | 0.82781653                   | 0.92954571                   | 0.97362811                   |
|           |            | -0.12083221 | -0.26450717 | -0.34265671                 | -0.31998660                  | -0.20351382                  | -0.02238574                  | 0.18836955                   | 0.39763774                   |
|           |            | 0.0498175   | 0.0436163   | -0.04560113                 | -0.18351806                  | -0.31683501                  | -0.40165863                  | -0.41475176                  | -0.35333620                  |
|           |            |             | 0.0465036   | 0.10970265                  | 0.13788809                   | 0.1008467                    | 6.68194092 $\times 10^{-4}$  | -0.13695355                  | -0.27710988                  |
|           |            |             | -0.01498699 | -0.00882680                 | 0.03892321                   | 0.11400345                   | 0.18207636                   | 0.21006834                   | 0.18012745                   |
|           |            |             |             | -0.01779187                 | -0.04466375                  | -0.05378245                  | -0.02456390                  | 0.043452675                  | 0.13160299                   |
|           |            |             |             | 4.71742793 $\times 10^{-3}$ | 7.83251152 $\times 10^{-4}$  | -0.02343994                  | -0.06235021                  | -0.09564726                  | -0.10096657                  |
|           |            |             |             |                             | 6.75606236 $\times 10^{-3}$  | 0.01774979                   | 0.01977216                   | 3.54892813 $\times 10^{-4}$  | -0.04165925                  |
|           |            |             |             |                             | -1.52353381 $\times 10^{-3}$ | 6.07514995 $\times 10^{-4}$  | 0.01236884                   | 0.03162417                   | 0.04696981                   |
|           |            |             |             |                             |                              | -2.54790472 $\times 10^{-3}$ | -6.88771926 $\times 10^{-3}$ | -6.67962023 $\times 10^{-3}$ | 5.10043697 $\times 10^{-3}$  |
|           |            |             |             |                             |                              | 5.00226853 $\times 10^{-4}$  | -5.54004549 $\times 10^{-4}$ | -6.05496058 $\times 10^{-3}$ | -0.01517900                  |
|           |            |             |             |                             |                              |                              | 9.55229711 $\times 10^{-4}$  | 2.61296728 $\times 10^{-3}$  | 1.97332536 $\times 10^{-3}$  |
|           |            |             |             |                             |                              |                              | -1.66137261 $\times 10^{-4}$ | 3.25814671 $\times 10^{-4}$  | 2.81768659 $\times 10^{-3}$  |
|           |            |             |             |                             |                              |                              |                              | -3.56329759 $\times 10^{-4}$ | -9.69947840 $\times 10^{-4}$ |
|           |            |             |             |                             |                              |                              |                              | 5.5645514 $\times 10^{-5}$   | -1.64709006 $\times 10^{-4}$ |
|           |            |             |             |                             |                              |                              |                              |                              | 1.32354367 $\times 10^{-4}$  |
|           |            |             |             |                             |                              |                              |                              |                              | -1.875841 $\times 10^{-5}$   |

Figura 1.2: Coeficientes ortogonales de Daubechies

los mismos valores para cada muestra. Por ello, a la hora de construir una Shapelet es necesario tener en cuenta la “forma” del patrón.

Comencemos por las primeras dos condiciones correspondientes a la propiedad de ortogonalidad que deben cumplir los filtros (ver ecuaciones 1.1 y 1.2). Esta propiedad es válida también para las Shapelets, pues es una condición necesaria para que un conjunto de vectores sea considerado un banco de filtros.

En el caso de la tercera condición en vez de considerar  $\frac{N}{2}$  ecuaciones se tomarán  $\frac{N}{2} - 2$  ecuaciones de momento nulo, la razón se debe a una cuarta condición que requieren las Shapelets. Suponga que se desea detectar un patrón  $m$  y la información que se conoce del mismo es que  $m(k) = m_k$  para toda  $k = 0, 1, \dots, N$ , las siguientes dos ecuaciones se conocen como ecuaciones de “*matching*”:

$$\sum_{k=0}^N v(k)m(k) = 0 \quad (1.4)$$

$$\sum_{k=0}^N v(k)m(k+1) = 0 \quad (1.5)$$

A diferencia de las ecuaciones usadas en DST-I, aquí se consideran dos ecuaciones de *matching* debido a las traslaciones diádicas que sufre el filtro en la fase de análisis de la señal, que de usar una sola podría provocar que no se detecte el patrón.

A continuación se muestra un ejemplo del método propuesto. Se desea detectar el patrón de la figura 1.3 en una señal de muestra, para ello es necesario determinar el banco de filtros de la Shapelet correspondiente al patrón en cuestión. Tomando  $N = 8$

se obtiene el siguiente sistema de ecuaciones no lineal:

$$v(0)^2 + v(1)^2 + v(2)^2 + v(3)^2 + v(4)^2 + v(5)^2 + v(6)^2 + v(7)^2 = 1$$

como condiciones de ortogonalidad las tres ecuaciones:

$$\begin{aligned} v(0)v(2) + v(1)v(3) + v(2)v(4) + v(3)v(5) + v(4)v(6) + v(5)v(7) &= 0 \\ v(0)v(4) + v(1)v(5) + v(2)v(6) + v(3)v(7) &= 0 \\ v(0)v(6) + v(1)v(7) &= 0 \end{aligned}$$

dos ecuaciones de momento nulo:

$$\begin{aligned} v(0) + v(1) + v(2) + v(3) + v(4) + v(5) + v(6) + v(7) &= 0 \\ v(1) + 2v(2) + 3v(3) + 4v(4) + 5v(5) + 6v(6) + 7v(7) &= 0 \end{aligned}$$

y dos ecuaciones de “*matching*”:

$$\begin{aligned} 0,2v(0) + 0,5v(1) + 0,45v(2) + 0,85v(3) + 0,8v(4) - 0,75v(5) + 0,25v(6) + 0,2v(7) &= 0 \\ 0,5v(0) + 0,45v(1) + 0,85v(2) + 0,8v(3) - 0,75v(4) + 0,25v(5) + 0,2v(6) + 0,55v(7) &= 0 \end{aligned}$$

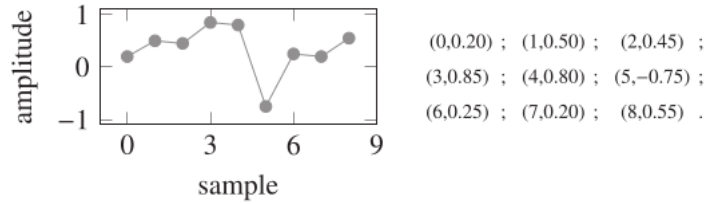


Figura 1.3: Ejemplo de una señal patrón ( $m$ ).

Tras aplicar métodos conocidos para la resolución de ecuaciones no lineales se obtiene como una de sus posibles soluciones:

$$v = (-0,0834, 0,1505, 0,5719, -0,7055, -0,0091, -0,2784, 0,2277, 0,1263)$$

Una vez obtenido  $v$  es posible calcular el valor del resto de los filtros  $u$ ,  $\tilde{v}$  y  $\tilde{u}$  que corresponden a los filtros de reconstrucción pasa bajo, de análisis pasa alto y análisis pasa bajo respectivamente, de donde:

$$\begin{aligned} u &= (-0,1263, 0,2277, 0,2784, -0,0091, 0,7055, 0,5719, -0,1505, -0,0834) \\ \tilde{v} &= (-0,0834, 0,1263, 0,2277, -0,2784, -0,0091, -0,7055, 0,5719, 0,1505) \\ \tilde{u} &= (0,1505, -0,5719, -0,7055, 0,0091, -0,2784, -0,2277, 0,1263, 0,0834) \end{aligned}$$

Al aplicar la transformada de shapelet usando los filtros anteriores se obtienen dos nuevos vectores  $cA$  y  $cD$  en la fase de análisis que constituyen los coeficientes wavelets de la representación de la señal original en la base  $\{R_{2k}u\}_{k=0}^{\frac{N}{2}-1} \cup \{R_{2k}v\}_{k=0}^{\frac{N}{2}-1}$

$$\left. \begin{aligned} D(z * \tilde{v}) &= cA \\ D(z * \tilde{u}) &= cD \end{aligned} \right\} \text{fase de análisis}$$

mientras que:

$$v * U(cA) + u * U(cD) = z \text{ fase de síntesis}$$

El nuevo vector  $cD$  es conocido en una dimensión como el vector de detalles y es donde se almacenarán los valores que se necesitan para determinar la presencia o no del patrón que describen  $u$  y  $v$  dentro de la señal. Para ello se define una función de medida de similitud  $\mathbb{S}$  tal que  $\mathbb{S} = e^{-|DST\_II(z)|^\alpha}$ , esta función enfatiza la presencia de ceros en el vector  $DST\_II(z)$  para un  $0 < \alpha < 1$ .

Sea la señal de muestra de la figura 1.4 definida para toda  $k \in \mathbb{Z}$  tal que  $0 \leq k \leq 63$ :

$$z(k) = \begin{cases} \cos\left(\frac{27\pi i}{8}\right) \sin\left(\frac{75\pi i}{8}\right), & \text{para } 0 \leq k \leq 40 \\ m(k), & \text{para } 41 \leq k \leq 49 \\ \cos\left(\frac{295\pi i}{32}\right) \sin\left(\frac{105\pi i}{32}\right), & \text{para } 50 \leq k \leq 63 \end{cases}$$

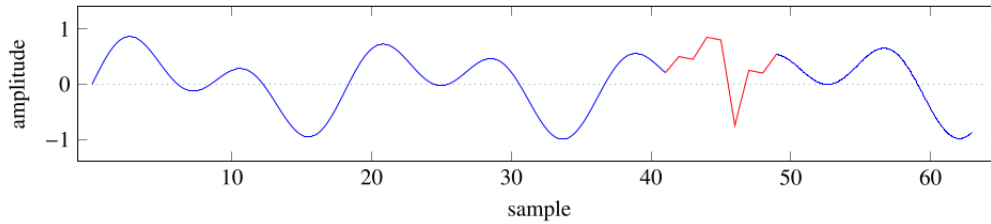


Figura 1.4: Señal de muestra con el patron  $m$  incluido.

y sean  $cA$  y  $cD$  los vectores de aproximación y detalles respectivamente obtenidos a partir de la DST-II, la figura 1.5 muestra los valores de estos vectores (naranja) y los valores de la medida de similitud (azul, con  $\alpha = 0,1$ ) aplicada a cada uno de los valores de los vectores  $cA$  y  $cD$ . Como se puede apreciar, para  $k = 53$  se obtiene un valor significativamente cercano a 1 en el gráfico de similitud, si se tiene en cuenta que, este valor corresponde al índice  $k = 21$  en el vector de aproximación  $cD$  y por tanto a la posición  $k = 42$  en la señal original, entonces se puede afirmar que el algoritmo detectó satisfactoriamente al patrón  $m$  en la señal de muestra.

Como parte del experimento, la medida  $\mathbb{S}$  fue aplicada a los resultados obtenidos para los vectores  $cA$  y  $cD$  tras aplicar la transformada de wavelet usando las Wavelets:

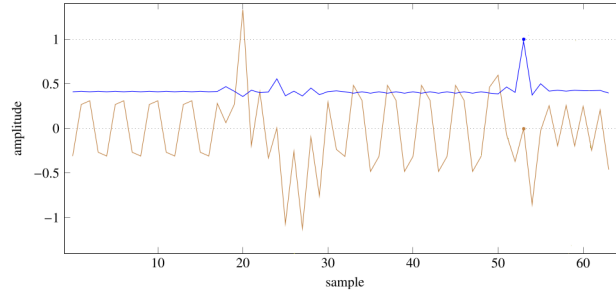


Figura 1.5: Representación de los vectores de aproximación (los primeros 32 valores) y detalle (los valores entre 32 y 63) y, la medida de similaridad con el patrón.

Haar, Daubechies 4, 6, 8, 10, 20, 30, 40, 50, 60 y 70, Symmlet 8 y 16, Coiflets 6, 12, 24, y 30, Beylkin 18 y por último Vaidyanathan 24. Sin embargo DST-II fue la única capaz de identificar al patrón  $m$  dentro de la señal  $z$ .

## 1.2. Wavelets en dos dimensiones

Hasta el momento se dispone de un método para identificar un patrón dado en una señal de muestra aplicando la transformada de shapelet. Dicho método funciona de manera efectiva en señales unidimensionales, ¿que pasaría entonces si agregamos una dimensión más?. Para poder extender el concepto Shapelet a dos dimensiones es necesario adaptar el conjunto de herramientas del cual se dispone en una dimensión a dos dimensiones tal que sea capaz de obtener similares resultados ya no solo para señales sino también para imágenes.

Anteriormente se dieron un conjunto de definiciones para mostrar un algoritmo que permitiera a partir de un vector construir un banco de filtros y por consiguiente una base wavelet. Una imagen puede ser representada por un vector en  $\ell^2(\mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2})$ , donde  $\ell^2(\mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2}) = \{z(i, j) | 0 \leq i \leq N_1 - 1, 0 \leq j \leq N_2 - 1\}$ , considerese entonces dicho espacio como el punto de partida.

### 1.2.1. Definiciones

Las siguientes definiciones fueron extraídas de [17].

**Definición 1.10** Para  $z \in \ell^2(\mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2})$  se define  $\hat{z}$  como la transformada discreta



de Fourier de  $z$  tal que:

$$\hat{z}(m_1, m_2) = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} z(n_1, n_2) e^{\frac{-2\pi i m_1 n_1}{N_1}} e^{\frac{-2\pi i m_2 n_2}{N_2}} \quad \forall m_1, m_2$$

**Definición 1.11** Para  $z \in \ell^2(\mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2})$  se define  $\tilde{z}$  como la reflexión conjugada de  $z$  tal que:

$$\tilde{z}(m_1, m_2) = \overline{z(-m_1, -m_2)} = \overline{z(N_1 - m_1, N_2 - m_2)} \quad \forall m_1, m_2$$

**Definición 1.12** Para todo  $w, z \in \ell^2(\mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2})$  se define la convolución entre ellos como  $z * w$  tal que:

$$(z * w)(m_1, m_2) = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} z(m_1 - n_1, m_2 - n_2) w(n_1, n_2) \quad \forall m_1, m_2$$

**Definición 1.13** Para  $z \in \ell^2(\mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2})$  y  $k_1, k_2 \in \mathbb{Z}$  se define  $R_{k_1, k_2} z$  como la traslación de  $z$  por  $k_1, k_2$  dado por:

$$R_{k_1, k_2} z(m_1, m_2) = z(m_1 - k_1, m_2 - k_2) \quad \forall m_1, m_2$$

### 1.2.2. Banco de filtros

**Teorema 1.1** (Ver demostración en [17]) Suponga que  $M_1, M_2 \in \mathbb{N}$ ,  $N_1 = 2M_1$ ,  $N_2 = 2M_2$  y  $u_1, u_2 \in \ell^2(\mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2})$ . Entonces para todo  $0 \leq k_1 \leq M_1 - 1$  y  $0 \leq k_2 \leq M_2 - 1$ ,  $\{R_{2k_1, 2k_2} u_1\}$  y  $\{R_{2k_1, 2k_2} u_2\}$  son conjuntos biortonormales si y solo si:

$$u^1 + u^2 + u^3 + u^4 = 4$$

donde

$$\begin{aligned} u^1 &= \hat{u}_1(n_1, n_2) \overline{\hat{u}_2(n_1, n_2)} \\ u^2 &= \hat{u}_1(n_1 + M_1, n_2) \overline{\hat{u}_2(n_1 + M_1, n_2)} \\ u^3 &= \hat{u}_1(n_1, n_2 + M_2) \overline{\hat{u}_2(n_1, n_2 + M_2)} \\ u^4 &= \hat{u}_1(n_1 + M_1, n_2 + M_2) \overline{\hat{u}_2(n_1 + M_1, n_2 + M_2)} \end{aligned}$$

**Teorema 1.2** (Ver demostración en [17]) Suponga que  $M_1, M_2 \in \mathbb{N}$ ,  $N_1 = 2M_1$ ,  $N_2 = 2M_2$  y  $u_i, v_i \in \ell^2(\mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2})$ , para toda  $i = 0, 1, 2, 3$ . Entonces para todo  $0 \leq k_1 \leq M_1 - 1$  y  $0 \leq k_2 \leq M_2 - 1$ ,

$$\bigcup_{m=0}^3 \{R_{2k_1, 2k_2} u_m\} \text{ y } \bigcup_{m=0}^3 \{R_{2k_1, 2k_2} v_m\}$$

son conjuntos biortogonales de  $\ell^2(\mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2})$  si y solo si:

$$A_u(n_1, n_2)^T \overline{A_v(n_1, n_2)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

para todo  $0 \leq n_1 \leq M_1 - 1$ ,  $0 \leq n_2 \leq M_2 - 1$ , donde  $A_u(n_1, n_2)$  es la matriz cuya  $m$ -ésima columna ( $m = 0, 1, 2, 3$ ) es:

$$\frac{1}{2} \begin{bmatrix} \hat{u}_m(n_1, n_2) \\ \hat{u}_m(n_1 + M_1, n_2) \\ \hat{u}_m(n_1, n_2 + M_2) \\ \hat{u}_m(n_1 + M_1, n_2 + M_2) \end{bmatrix}$$

y  $\tilde{A}(n_1, n_2)$  es la matriz cuya  $m$ -ésima columna ( $m = 0, 1, 2, 3$ ) es:

$$\frac{1}{2} \begin{bmatrix} \hat{v}_m(n_1, n_2) \\ \hat{v}_m(n_1 + M_1, n_2) \\ \hat{v}_m(n_1, n_2 + M_2) \\ \hat{v}_m(n_1 + M_1, n_2 + M_2) \end{bmatrix}$$

**Definición 1.14** Suponga que  $M_1, M_2 \in \mathbb{N}$ ,  $N_1 = 2M_1$  y  $N_2 = 2M_2$ . Se definen los operadores

$$D^l : \ell^2(\mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2}) \rightarrow \ell^2(\mathbb{Z}_{\frac{N_1}{2^l}} \times \mathbb{Z}_{\frac{N_2}{2^l}})$$

de submuestreo y

$$U^l : \ell^2(\mathbb{Z}_{\frac{N_1}{2^l}} \times \mathbb{Z}_{\frac{N_2}{2^l}}) \rightarrow \ell^2(\mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2})$$

de sobremuestreo como:

$$D^l(z)(n_1, n_2) = z(2^l n_1, 2^l n_2) \quad \forall z \in \ell^2(\mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2})$$

y

$$U^l(z)(n_1, n_2) = \begin{cases} z\left(\frac{n_1}{2^l}, \frac{n_2}{2^l}\right), & 2^l | n_1, 2^l | n_2 \\ 0, & \text{en otro caso} \end{cases}$$

$$\forall z \in \ell^2(\mathbb{Z}_{\frac{N_1}{2^l}} \times \mathbb{Z}_{\frac{N_2}{2^l}})$$

donde  $D^1 = D$ ,  $U^1 = U$  y  $D^l(z) = (D \circ D^{l-1})(z)$ ,  $U^l(z) = (U \circ U^{l-1})(z)$  para  $l > 1$ .

**Teorema 1.3** (Ver demostración en [17]) Suponga que  $M_1, M_2 \in \mathbb{N}$ ,  $N_1 = 2M_1$ ,  $N_2 = 2M_2$  y  $u_0, u_1, u_2, u_3, s_0, s_1, s_2, s_3 \in \ell^2(\mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2})$ . Sea la matriz  $A(n_1, n_2)$  definida como en el Teorema 1.2, entonces se tiene una reconstrucción perfecta:

$$\sum_{i=0}^3 \tilde{s}_i * U(D(z * \tilde{u}_i)) = z$$

para toda  $z \in \ell^2(\mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2})$ , si y solo si:

$$A(n_1, n_2) \begin{bmatrix} \hat{s}_0 \\ \hat{s}_1 \\ \hat{s}_2 \\ \hat{s}_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

para todo  $n_1, n_2$ . En el caso de que  $A(n_1, n_2)$  sea unitaria, entonces  $s_i = \tilde{u}_i$  para  $i = 0, 1, 2, 3$ .

Analizemos que resultados podemos obtener a partir de las definiciones y teoremas anteriores. De los teoremas 1.2 y 1.3 se sabe que si:

$$\bigcup_{m=0}^3 \{R_{2k_1, 2k_2} u_m\}$$

es un conjunto biortogonal consigo mismo en  $\ell^2(\mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2})$  entonces la matriz  $A(n_1, n_2)$  es unitaria y por tanto:

$$\sum_{i=0}^3 u_i * U(D(z * \tilde{u}_i)) = z$$

constituye una reconstrucción perfecta de una imagen  $z$ . Esto quiere decir que al igual que en una sola dimensión, el problema de crear una base wavelet que permita detectar un patrón, se reduce al cálculo efectivo de un banco de filtros  $\{u_i\} \cup \{\tilde{u}_i\}$  ahora biortogonal. Este constituye el próximo paso en la tarea de crear una Shapelet de dos dimensiones.

# Capítulo 2

## Propuesta

Hasta el momento se conoce como crear un banco de filtros en  $\ell^2(\mathbb{Z}_N)$  tal que la DST-II sea capaz de reconocer un patrón en una señal unidimensional. Tal como se vio en la sección 1.2, una imagen puede ser definida como un vector en  $\ell^2(\mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2})$  al igual que un patrón. Si bien, en el estudio de la transformada de Shapelet de una dimensión era posible construir un filtro que reconociese un patrón dado, dicha tarea también es posible realizarla en dos dimensiones, aunque para ello sea necesario realizar algunos ajustes y consideraciones extras.

### 2.1. Definición del filtro

Como bien se sabe, para determinar el banco de filtros de la Shapelet se debe resolver primeramente un sistema no lineal cuyas variables son las componentes del vector que se usará para construir la base wavelet. Dicho vector pertenece al espacio  $\ell^2(\mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2})$  por lo que contiene un total de  $N_1 N_2$  componentes. A la hora de resolver el sistema de ecuaciones se estaría tratando con un sistema sumamente costoso para los equipos de cómputos actuales. Por tanto, es necesario encontrar una alternativa para obtener dicho vector.

Según se analizó anteriormente, la resolución del sistema de ecuaciones no lineal que se obtiene tratando con vectores en una dimensión es más factible que la de dos dimensiones; entonces, ¿sería posible construir un vector en  $\ell^2(\mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2})$  a partir de un vector en  $\ell^2(\mathbb{Z}_N)$ ?

Suponga que  $N$  es un entero positivo par y que se tienen dos vectores  $v_0, v_1 \in \ell^2(\mathbb{Z}_N)$  tal que  $\{R_{2k}v_0\}_{k=0}^{\frac{N}{2}-1} \cup \{R_{2k}v_1\}_{k=0}^{\frac{N}{2}-1}$  constituye una base ortonormal de  $\ell^2(\mathbb{Z}_N)$  y cuatro

vectores  $w_{0,0}, w_{0,1}, w_{1,0}, w_{1,1} \in \ell^2(\mathbb{Z}_N \times \mathbb{Z}_N)$  tal que:

$$\begin{aligned} w_{0,0}(n_1, n_2) &= v_0(n_1)v_0(n_2) \\ w_{0,1}(n_1, n_2) &= v_0(n_1)v_1(n_2) \\ w_{1,0}(n_1, n_2) &= v_1(n_1)v_0(n_2) \\ w_{1,1}(n_1, n_2) &= v_1(n_1)v_1(n_2) \end{aligned} \quad (2.1)$$

para todo  $w_{i,j}$  con  $i, j \in \{0, 1\}$  entonces se cumple que:

$$\begin{aligned} \hat{w}_{i,j}(m_1, m_2) &= \sum_{n_1=0}^{N-1} \sum_{n_2=0}^{N-1} w_{i,j}(n_1, n_2) e^{\frac{-2\pi i m_1 n_1}{N}} e^{\frac{-2\pi i m_2 n_2}{N}} \\ &= \sum_{n_1=0}^{N-1} \sum_{n_2=0}^{N-1} v_i(n_1) v_j(n_2) e^{\frac{-2\pi i m_1 n_1}{N}} e^{\frac{-2\pi i m_2 n_2}{N}} \\ &= \left( \sum_{n_1=0}^{N-1} v_i(n_1) e^{\frac{-2\pi i m_1 n_1}{N}} \right) \left( \sum_{n_2=0}^{N-1} v_j(n_2) e^{\frac{-2\pi i m_2 n_2}{N}} \right) \\ \hat{w}_{i,j}(m_1, m_2) &= \hat{v}_i(m_1) \hat{v}_j(m_2) \end{aligned}$$

con lo cual se tiene una expresión para  $\hat{w}_{i,j}$  en función de los vectores  $v_i$  y  $v_j$  que lo componen. Considere ahora el teorema 1.1, tal que  $u^1, u^2, u^3$  y  $u^4$  se definen como:

$$\begin{aligned} u^1 &= \hat{w}_{i_1, j_1}(m_1, m_2) \overline{\hat{w}_{i_2, j_2}(m_1, m_2)} = \hat{v}_{i_1}(m_1) \hat{v}_{j_1}(m_2) \overline{\hat{v}_{i_2}(m_1) \hat{v}_{j_2}(m_2)} \\ u^2 &= \hat{w}_{i_1, j_1}\left(m_1, m_2 + \frac{N}{2}\right) \overline{\hat{w}_{i_2, j_2}\left(m_1, m_2 + \frac{N}{2}\right)} \\ &= \hat{v}_{i_1}(m_1) \hat{v}_{j_1}\left(m_2 + \frac{N}{2}\right) \overline{\hat{v}_{i_2}(m_1) \hat{v}_{j_2}\left(m_2 + \frac{N}{2}\right)} \\ u^3 &= \hat{w}_{i_1, j_1}\left(m_1 + \frac{N}{2}, m_2\right) \overline{\hat{w}_{i_2, j_2}\left(m_1 + \frac{N}{2}, m_2\right)} \\ &= \hat{v}_{i_1}\left(m_1 + \frac{N}{2}\right) \hat{v}_{j_1}(m_2) \overline{\hat{v}_{i_2}\left(m_1 + \frac{N}{2}\right) \hat{v}_{j_2}(m_2)} \\ u^4 &= \hat{w}_{i_1, j_1}\left(m_1 + \frac{N}{2}, m_2 + \frac{N}{2}\right) \overline{\hat{w}_{i_2, j_2}\left(m_1 + \frac{N}{2}, m_2 + \frac{N}{2}\right)} \\ &= \hat{v}_{i_1}\left(m_1 + \frac{N}{2}\right) \hat{v}_{j_1}\left(m_2 + \frac{N}{2}\right) \overline{\hat{v}_{i_2}\left(m_1 + \frac{N}{2}\right) \hat{v}_{j_2}\left(m_2 + \frac{N}{2}\right)} \end{aligned}$$

Sea  $s = u^1 + u^2 + u^3 + u^4$ , hallando factor común  $\hat{v}_{i_1}(m_1) \overline{\hat{v}_{i_2}(m_1)}$  y  $\hat{v}_{i_1}\left(m_1 + \frac{N}{2}\right) \overline{\hat{v}_{i_2}\left(m_1 + \frac{N}{2}\right)}$  se tiene:

$$s = \left( \hat{v}_{i_1}(m_1) \overline{\hat{v}_{i_2}(m_1)} + \hat{v}_{i_1}\left(m_1 + \frac{N}{2}\right) \overline{\hat{v}_{i_2}\left(m_1 + \frac{N}{2}\right)} \right) \left( \hat{v}_{j_1}(m_2) \overline{\hat{v}_{j_2}(m_2)} + \hat{v}_{j_1}\left(m_2 + \frac{N}{2}\right) \overline{\hat{v}_{j_2}\left(m_2 + \frac{N}{2}\right)} \right)$$

Como  $\{R_{2k}v_0\}_{k=0}^{\frac{N}{2}-1} \cup \{R_{2k}v_1\}_{k=0}^{\frac{N}{2}-1}$  constituye una base ortonormal de  $\ell^2(\mathbb{Z}_N)$ , se cumple que la matriz:

$$\frac{1}{\sqrt{2}} \begin{bmatrix} \hat{v}_0(n) & \hat{v}_1(n) \\ \hat{v}_0\left(n + \frac{N}{2}\right) & \hat{v}_1\left(n + \frac{N}{2}\right) \end{bmatrix}$$

es unitaria para toda  $n$ . Esto quiere decir que:

$$\hat{v}_i(n)\overline{\hat{v}_j(n)} + \hat{v}_i\left(n + \frac{N}{2}\right)\overline{\hat{v}_j\left(n + \frac{N}{2}\right)} = \begin{cases} 2, & \text{si } i = j \\ 0, & \text{si } i \neq j \end{cases}$$

por lo tanto:

$$s = \begin{cases} 4, & \text{si } i_1 = i_2 \text{ y } j_1 = j_2 \\ 0, & \text{en caso contrario} \end{cases}$$

y el conjunto  $\bigcup_{i,j \in \{0,1\}} \{R_{2k_1, 2k_2}w_{i,j}\}_{k_1=0, k_2=0}^{\frac{N}{2}-1, \frac{N}{2}-1}$  constituye un conjunto biortogonal en  $\ell^2(\mathbb{Z}_N \times \mathbb{Z}_N)$ . Este último resultado trae como implicación que:

$$\sum_{i=0}^1 \sum_{j=0}^1 w_{i,j} * U(D(z * \tilde{w}_{i,j})) = z$$

constituya una reconstrucción perfecta de  $z$ . Volviendo entonces a la interrogante de si era posible construir un filtro de dos dimensiones a partir de un vector de una dimensión, si se define el vector de dos dimensiones como en 2.1 se obtiene el filtro deseado. La única condición que debe cumplir es que  $N_1 = N_2 = N$ .

## 2.2. Cálculo del filtro

Nuestro problema se reduce entonces a determinar un conjunto de vectores  $v_0, v_1 \in \ell^2(\mathbb{Z}_N)$  tal que  $\{R_{2k}v_0\}_{k=0}^{\frac{N}{2}-1} \cup \{R_{2k}v_1\}_{k=0}^{\frac{N}{2}-1}$  constituya una base ortonormal en dicho espacio. Sin embargo en la sección 1.1.3 se brinda un algoritmo para hallar dichos vectores a partir de un patrón de muestra. Por tanto, nuestro problema de detectar un patrón en dos dimensiones se reduce al problema de detectar un patrón en una dimensión. Analizemos nuevamente las condiciones que deben cumplirse para el cálculo del filtro.

La primera condición que debía cumplir  $v$  era la condición de energía unitaria (vea ecuación 1.1), mientras que la segunda condición era la de ortogonalidad (ver

ecuación 1.2). La tercera y cuarta condiciones representaban el momento nulo y las ecuaciones de *matching* respectivamente. Detengámonos en estas últimas.

Cuando se proponen las ecuaciones de *matching* en una dimensión son necesarias dos ecuaciones, esto se debía a que las traslaciones diádicas mantienen la paridad del índice inicial tomado en la señal de prueba para realizar la convolución con el filtro, y si el patrón estuviese insertado en la señal pero en una posición con paridad distinta, entonces no hubiese sido posible detectarlo. Para resolver este problema se utilizan dos ecuaciones en vez de una. Para el caso de dos dimensiones se puede presentar el mismo problema, sin embargo ya no solo ocurriría al desplazar el índice de las columnas sino también el de las filas, por ello serían necesario utilizar cuatro ecuaciones de *matching* para el proceso de determinación del filtro.

Al añadir dos nuevas ecuaciones de *matching* el sistema se puede volver incompatible, para evitarlo se eliminan dos ecuaciones más de momento nulo. De esta forma, la tercera condición la constituyen  $\frac{N}{2} - 4$  ecuaciones de desvanecimiento:

$$\sum_{k=0}^{N-1} v(k)k^b = 0, \quad \forall b = 0, 1, \dots, \frac{N}{2} - 5$$

mientras que la cuarta condición está dada por las siguientes ecuaciones de *matching*:

$$\begin{aligned} \sum_{n_1=0}^{N-1} \sum_{n_2=0}^{N-1} v(n_1)v(n_2)m(n_1, n_2) &= 0 \\ \sum_{n_1=0}^{N-1} \sum_{n_2=0}^{N-1} v(n_1)v(n_2)m(n_1, n_2 + 1) &= 0 \\ \sum_{n_1=0}^{N-1} \sum_{n_2=0}^{N-1} v(n_1)v(n_2)m(n_1 + 1, n_2) &= 0 \\ \sum_{n_1=0}^{N-1} \sum_{n_2=0}^{N-1} v(n_1)v(n_2)m(n_1 + 1, n_2 + 1) &= 0 \end{aligned}$$

donde  $m$  representa un patrón de  $\ell^2(\mathbb{Z}_{N+1} \times \mathbb{Z}_{N+1})$ .

Como se puede apreciar, para el cálculo del filtro es necesario resolver un sistema de ecuaciones no lineales de  $N$  variables con  $N$  ecuaciones al igual que en el caso de una dimensión. Lo cual constituye una reducción significativa del tiempo teórico que se tenía anteriormente.

Una vez resuelto el sistema aplicando algún método de resolución de sistemas de ecuaciones, se obtienen los valores de  $v$  con los cuales se construyen los filtros en dos dimensiones y se aplica la DST\_II. Para ello se realiza la fase de análisis de la imagen,

obteniéndose 4 nuevas imágenes  $cA$ ,  $cH$ ,  $cV$  y  $cD$  llamadas imagen de aproximación (obtenida con  $D(z * w_{1,1})$ ), matriz de detalles horizontales (obtenida con  $D(z * w_{0,1})$ ), matriz de detalles verticales (obtenida con  $D(z * w_{1,0})$ ) y matriz de detalles (obtenida con  $D(z * w_{0,0})$ ) respectivamente.

Al igual que en el caso de una dimensión. El valor que indica la presencia del patrón se encuentra en la matriz de detalles  $cD$  por lo que aplicando la medida de similitud propuesta en la sección 1.1.3 se selecciona el índice más cercano a 1 como la localización del patrón en la imagen.

Si bien hasta ahora solo se tiene una idea teórica del funcionamiento del algoritmo a continuación se presenta la implementación del mismo en correspondencia con resultados obtenidos durante los experimentos realizados.



## Capítulo 3

# Detalles de Implementación y Experimentos

La implementación del algoritmo de reconocimiento de patrones propuesto se realizó en el lenguaje de programación python. La razón principal: el conjunto de bibliotecas que incluye tanto para la resolución del sistema de ecuaciones, como para determinar la transformada de wavelet de un vector lo mismo en  $\ell^2(\mathbb{Z}_N)$  que en  $\ell^2(\mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2})$ .

El programa se descompone en dos algoritmos y cada algoritmo en dos partes, es decir: un primer algoritmo para la detección de patrones unidimensionales en una señal y un segundo para detectar un patrón de dimensiones cuadradas en una imagen o mamografía. En la primera fase de cada algoritmo se plantea el sistema de ecuaciones cuya solución es el filtro pasa alto y se le da solución empleando el método numérico que se mostrará a continuación, mientras que en la segunda fase, se toma la wavelets creada a partir del banco de filtro con el vector hallado y se determina la transformada de la imagen de muestra (la cual no necesita ser cuadrada), tras la cual se obtienen las cuatro matrices o imágenes mencionadas anteriormente; de ellas se analiza  $cD$  en busca de valores cercanos a cero con la función de similaridad  $\mathbb{S}$  y se seleccionan los mejores candidatos de ocurrencia del patrón.

Como pudo haber notado, el problema principal del algoritmo es la resolución del sistema de ecuaciones no lineales. Por lo que se analizará primero este punto antes de entrar en la implementación del programa en general.

### 3.1. Métodos para resolver sistemas de ecuaciones

Para la resolución del sistema no lineal, se consideraron varios métodos numéricos tales como: Newton, Cuasi-Newton, Descenso por gradiente, Runge Kutta y otro empleado para la resolución de modelos de optimización. Analicemos cada uno de ellos y determinemos cual produciría un resultado más cercano al que se desea obtener.

#### 3.1.1. Método de Newton

En el caso del método de Newton, consiste en un algoritmo iterativo, tal que en cada iteración se actualiza el valor  $x_0$  inicial y el jacobiano del sistema, a fin de que  $x_0$  siempre se encuentre en una vecindad cada vez más reducida de la raíz del sistema. Este método es muy efectivo en pequeños sistemas de ecuaciones y converge rápidamente en caso de que se tome un valor inicial relativamente cercano a la raíz. Sin embargo, nuestro problema no entra dentro de dichos sistemas de ecuaciones, ya que la dimensión de los vectores que se usarán para el filtro pueden ser bastante grandes por lo que la posibilidad de encontrar un primer valor  $x_0$  cercano a la raíz se hace muy pequeña y en muchos casos el algoritmo podría no converger.

La figura 3.1 muestra la efectividad del método de Newton para distintas dimensiones del sistema de ecuaciones.

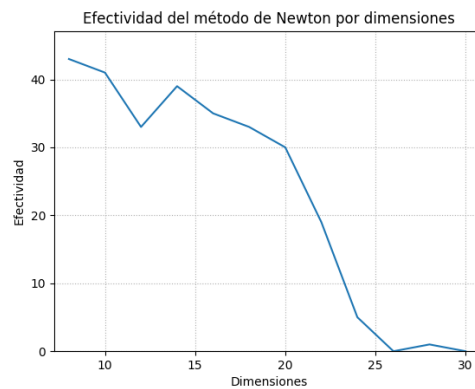


Figura 3.1: Efectividad (%) del método de Newton en el modelo para un aumento de la dimensión.

Durante el experimento, se ejecutó el método 100 veces por cada dimensión y se determinó el porcentaje que representa el total de casos satisfactorios. Como se puede apreciar, con este método se pueden obtener valores correctos hasta con sistemas de dimensión  $N = 26$ , aunque para estos últimos valores la posibilidad de encontrarlos es muy baja.

### 3.1.2. Método de Cuasi Newton

Por otro lado, el método de Cuasi-Newton constituye una mejora del método de Newton en el sentido de que no es necesario hallar en cada iteración la inversa del jacobiano del sistema, sino que utiliza una matriz definida positiva para aproximar la inversa de la matriz hessiana, simplificando de esta forma el proceso. Sin embargo, padece del mismo mal que el método de Newton, pues requiere un valor inicial  $x_0$  cercano a la raíz para converger.

Usando el mismo experimento que en el método de Newton, Se tomaron 100 patrones aleatorios para cada dimensión y se determino la efectividad del método Cuasi-Newton Broyden Fletcher Goldfarb Shanno (BFGS). La figura 3.2 muestra los resultados del análisis.

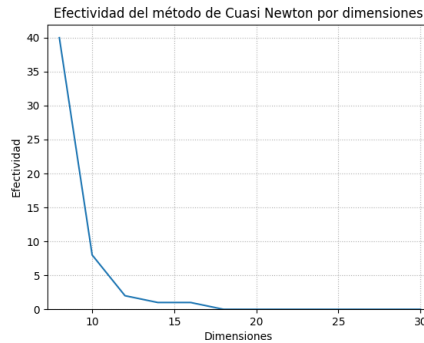


Figura 3.2: Efectividad (%) del método de Cuasi Newton en el modelo para un aumento de la dimensión.

Como puede apreciar, inicialmente tiene una efectividad del 40% para un sistema de dimensión 8 sin embargo con un aumento de esta, comienza a reducir hasta que, a partir de  $N = 18$  se vuelve inefectivo.

### 3.1.3. Método de Descenso por Gradiente

El método del Descenso por gradiente es un método usado para la determinación de mínimos, por lo tanto para usar este método fue necesario trabajar con la norma de del vector formado por las ecuaciones del sistema, de esta forma el valor  $x$  que minimice la norma constituye la raíz en el sistema original. La idea de optimización de este método, es usar la dirección contraria al gradiente en el punto actual como dirección de búsqueda, moverse una distancia  $\alpha$  y volver a repetir la operación. A medida que vaya acercándose al mínimo, la distancia  $\alpha$  se reduce y más lenta es la convergencia. El problema fundamental de este método es que puede converger a mínimos locales,

muy lejos de la raíz verdadera del sistema y esto podría traer dificultades a la hora de detectar el patrón.

La figura 3.3 muestra la efectividad de este método para 12 valores de dimensiones distintas.

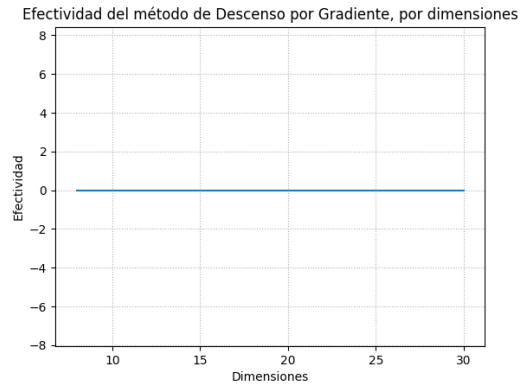


Figura 3.3: Efectividad (%) del método de Descenso por Gradiente en el modelo para un aumento de la dimensión.

Aunque al compararlo con los primeros métodos vistos, rápidamente sería descartado, puede ser útil como un método de aproximación inicial.

### 3.1.4. Método de Runge Kutta

El próximo método es el método de Runge Kutta. Este es un método numérico que al igual que el método de Newton, usa el jacobiano del sistema. Más aún, los métodos de Runge Kutta son generalizaciones de la fórmula de Euler:

$$x_{i+1} = x_i + hf(t_i, x_i)$$

donde el valor de  $f$  es reemplazado por un promedio ponderado de valores de  $f$  en el intervalo  $t_i \leq t \leq t_{i+1}$ . Es decir:

$$x_{i+1} = x_i + h(w_1k_1 + w_2k_2 + \dots + w_mk_m)$$

En este caso se usará Runge Kutta de orden 4. Realizado el mismo experimento que con los métodos anteriores se obtiene el resultado mostrado en la figura 3.4.

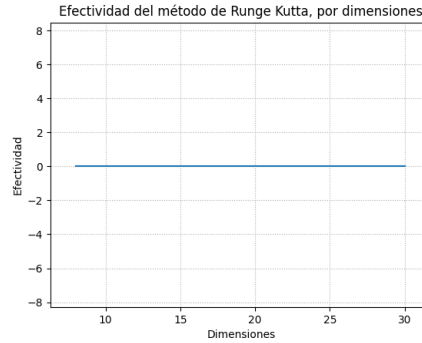


Figura 3.4: Efectividad (%) del método de RungeKutta en el modelo para un aumento de la dimensión.

### 3.1.5. Gekko

El ultimo método que se analizará constituye un método usado para la resolución de problemas de maximización y minimización de funciones. Como parte de las bibliotecas de python, se encuentra *gekko*[11]. *Gekko* es un paquete de optimización y *machine learning* que usa diferenciación automática y algoritmos basados en gradientes como APOPT o IPOPT para encontrar solución a problemas de optimización. Si cada ecuación constituye una restricción y se considera la norma de las restricciones como la función objetivo, entonces es posible transformar el sistema de ecuaciones no lineales en un modelo de optimización no lineal. Al usar *gekko* entonces para resolver los sistemas con distintas dimensiones, se obtienen los datos mostrados en la figura 3.5.

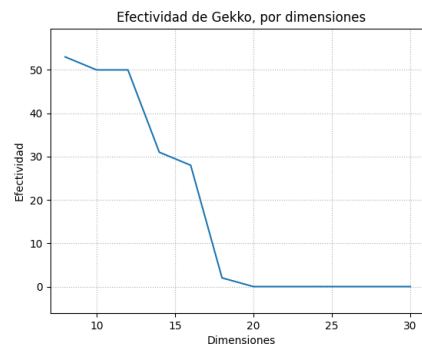


Figura 3.5: Efectividad (%) del método de Gekko en el modelo para un aumento de la dimensión.

En comparación con los primeros dos métodos, aunque a partir de un valor de

dimensión dado se vuelve ineficiente, muestra un comportamiento con valores de efectividad muy buenos en comparación con el método de Cuasi Newton pero inferiores a los de Newton.

### 3.1.6. Mejores Resultados

Hasta el momento se han analizado los métodos por separado, de los cuales el método de Newton a dado los mejores resultados. Tal como se mencionó varios de ellos dependen de una buena aproximación inicial. Dado que el método del descenso por gradiente y Runge Kutta no entran dentro de estos, se usarán para lograr una buena aproximación inicial y posteriormente aplicar alguno de los otros métodos propuestos. Realizando nuevamente el experimento usando cada método de aproximación inicial con cada método para encontrar la raíz del sistema, se obtienen los resultados mostrados en la figura 3.6.

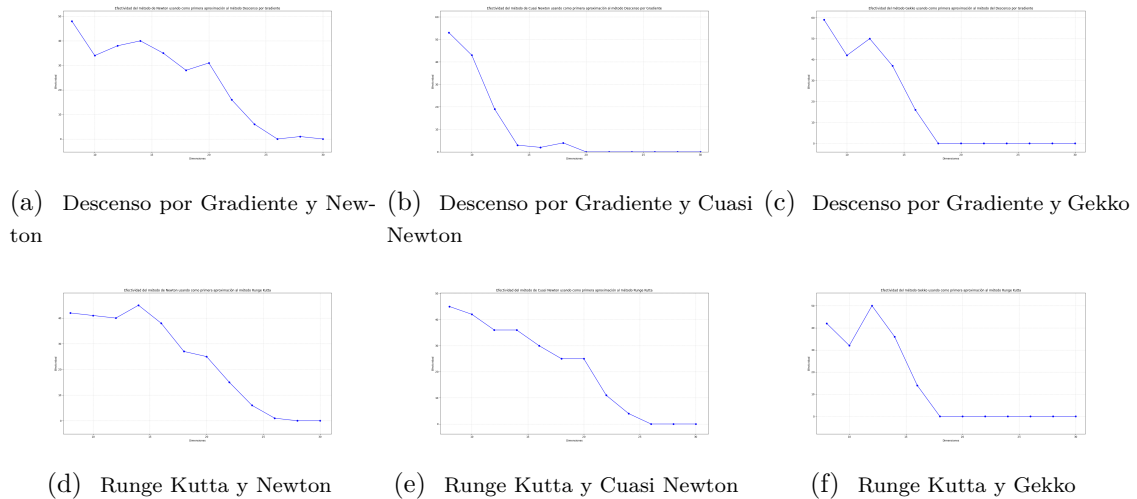


Figura 3.6: Resultados obtenidos empleando los metodos del Descenso por Gradiente y Runge Kutta para encontrar una aproximación inicial y, los métodos Newton, Cuasi Newton y Gekko para resolver el sistema.

Segun los resultados mostrados Newton, usando como aproximación inicial a Runge Kutta constituye la combinación de métodos con mayor compatibilidad con el sistema propuesto y por consiguiente, la combinación que se usará para determinar los filtros de las Shapelets, al menos para el caso de una dimensión.

## 3.2. Algoritmo de reconocimiento en una dimensión

Una vez se tiene la señal del patrón, nuestro primer paso es definir el sistema de ecuaciones a partir del mismo y resolverlo aplicando los métodos de Runge Kutta y Newton. Al resolver el sistema se obtienen los valores de las componentes del primer vector (llamémoslo  $v$ ). A partir de  $v$  se construyen los vectores  $u$ ,  $\tilde{u}$  y  $\tilde{v}$  siguiendo las definiciones dadas en la sección 1.1.1. Con estos cuatro vectores se construye el banco de filtros y con él la wavelet. Para ello se emplea la biblioteca *pywt* de python, con la cual es posible crear una Wavelet recibiendo como parámetro el banco de filtros[16](ver código 3.1):

```
1 shapelet = pywt.Wavelet(filter_bank = [u_, v_, u, v])
```

Ejemplo de código 3.1: Creación de una wavelet a partir del banco de filtros

El siguiente paso es tomar una señal en la cual se quiere determinar si el patrón está o no presente y aplicar la transformada de wavelet usando la wavelet creada con el filtro:

```
1 cA, cD = pywt.dwt(signal, shapelet)
```

Ejemplo de código 3.2: Cálculo de la transformada discreta de Wavelet usando *pywt*

para posteriormente determinar según los valores cercanos a cero de  $cD$  la presencia del patrón en la señal.

Considere el caso de ejemplo de la sección 1.1.3:

$$m = (0,2,0,5,0,45,0,85,0,8,-0,75,0,25,0,2,0,55)$$

Tras construir el sistema y resolverlo se obtiene como una de las posibles soluciones:

$$v = (0,0445,0,4373,0,1074,-1,6020,1,0664,-0,2296,0,1959,-0,0199)$$

mientras que:

$$u = (0,4373,-0,0445,-0,0199,0,1959,-0,2296,-1,0664,-1,6020,-0,1074)$$

$$\tilde{v} = (0,0445,-0,0199,0,1959,-0,2296,1,0664,-1,6020,0,1074,0,4373)$$

$$\tilde{u} = (0,4373,-0,1074,-1,6020,-1,0664,-0,2296,-0,1959,-0,0199,-0,0445)$$

Note que los valores obtenidos son completamente distintos a los del ejemplo de la sección 1.1.3, sin embargo al comprobar las condiciones de la 1.1-1.5 las cumple todas. Con los cuatro vectores anteriores se crea el banco de filtros  $[\tilde{u}, \tilde{v}, u, v]$  y se construye la Shapelet.

Al procesar la señal de muestra definida también en el ejemplo anterior, se obtiene el gráfico de valores de  $cD$  mostrado en la figura 3.7, en el cuál se puede apreciar que el pico que indica la detección del patrón se encuentra en la posición 24.

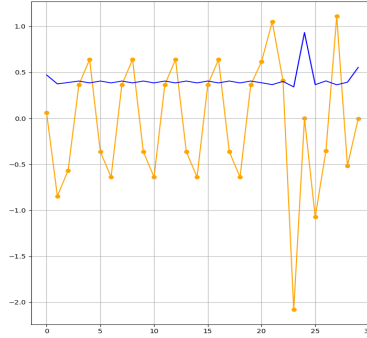


Figura 3.7: Vector  $cD$  obtenido tras aplicar la DST (naranja) y, medida de similaridad con  $\alpha = 0,1$  (azul).

Como se cumple que:

$$\begin{aligned}
 cD(n) = 0 &\Leftrightarrow D(f * \tilde{v})(n) = 0 \\
 &\Leftrightarrow (f * \tilde{v})(2n) = 0 \\
 &\Leftrightarrow \sum_{m=0}^{N-1} v(m)f(2n - m) = 0
 \end{aligned}$$

entonces, si  $cD(n) = 0$  el patrón se encuentra contenido en  $f$  en el intervalo  $[2n - N + 1, 2n]$  por tanto tomando  $n = 24$ , se cumple que  $2n - N + 1 = 41$  que es la posición exacta en la que comienza el patrón dentro de la señal.

Como parte del análisis del programa se generaron varios patrones de forma aleatoria con dimensión también variable y se insertaron dentro de una señal en una posición *random* con una probabilidad de 0,5. Sin embargo fue detectado del patrón satisfactoriamente en el 100% de los casos que aparecía al igual que la detección de la ausencia en el 100% de los casos en los que no se incluyó el patrón en la señal.

Como un segundo experimento Se generaron 676 casos de prueba con patrón aleatorio e inclusión aleatoria en la señal, pero esta vez con un ruido en el orden  $10^{-3}$ . La figura 3.8 muestra los gráficos de precisión y recobrado del algoritmo durante el



proceso de detección de la presencia o no del patrón en dependencia del umbral usado para la medida de similitud.

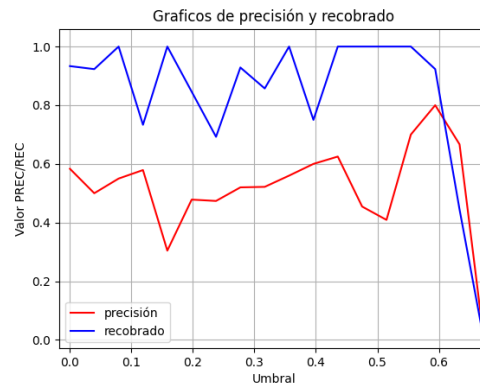


Figura 3.8: Precisión y Recobrado del algoritmo de una dimensión para distintos valores del umbral usando la mediada  $\mathcal{S}$ .

Note que aun con la presencia de ruido en el patrón dentro de la señal, se muestra un valor de precisión y recobrado bastante aceptable. Como un último experimento se usaron otras bases wavelets (las mencionadas al final de la sección 1.1.3) para intentar detectar el patrón sin éxito alguno. La razón de esto se debe a que ninguna de estas wavelets tiene en cuenta el patrón que se desea localizar por tanto no se puede esperar obtener un comportamiento distinto para distintos casos de patrones que se incluyan en una señal.

### 3.3. Algoritmo de reconocimiento en dos dimensiones

Nuestro problema principal es crear una base wavelet que permita reconocer anomalías en una mamografía digital. Para ello tanto la mamografía como la anomalía son representadas como imágenes con la particularidad de que esta última debe tener iguales proporciones de ancho y altura. Estas imágenes que representan los datos a procesar se encuentran en formato `.dcm`, también conocido como imágenes DICOM. Este es un formato internacional que se desarrolló con el fin de almacenar las imágenes del cuerpo tomadas con equipos de imágenes médicas. Incluye imágenes de resonancia magnética y tomografía por ordenador, imágenes de ultrasonido, mamografías, entre otras. DCM también almacena la información del paciente en los metadatos, que forman parte de un único archivo.

En el software desarrollado se emplea la biblioteca de python *pydicom*[15], creada específicamente para el trabajo con archivos de tipo *dcm*. Suponga que se desea leer primeramente la imagen de la anomalía que a su vez constituye el patrón, usando *pydicom* sería tan simple como ejecutar:

```
1 ds = pydicom.dcmread(path, force=True)
```

Ejemplo de código 3.3: Leer una imagen *.dcm*

con lo cual se lee la imagen de la dirección *path* y se almacena en *ds* como un archivo de tipo *pydicom.FileDataset*. De aquí lo que nos interesa es el valor de *ds.pixel\_array*, que contiene la matriz de valores de los píxeles de la mamografía.

El siguiente paso, es crear el sistema de ecuaciones no lineales. Y aquí una nota muy importante, aunque el método de Newton usado con Runge Kutta mostraron mejores resultados en el caso de una dimensión, al incluir dos ecuaciones nuevas con una cantidad de  $N^2$  términos, el método de Newton comienza a presentar dificultades para resolver el sistema, mientras que por otro lado, Gekko muestra un buen desempeño con el aumento de términos, esto se debe a que dicha biblioteca posee su propio método de representar los datos. Entonces, dado que ambos métodos (Gekko y Runge Kutta) requieren una representación distinta de los datos, existen dos formulaciones del modelo en el programa. El primero de ellos (Runge Kutta) usa la función multivariable como la función a la cual se le quiere hallar la raíz y el jacobiano del sistema. Para esto se crearon dos funciones: *func* y *jac* que retornan la evaluación de la función multivariable y el jacobiano en un punto dado respectivamente. Para mejorar la eficiencia del programa, se planteó el sistema en forma matricial usando la biblioteca de Python: *numpy*, de esta forma evaluar cada función en un valor del dominio se transforma en operaciones entre matrices *numpy*, que se realizan en un tiempo mucho menor en comparación a las operaciones con matrices en python.

Por su parte, Gekko, usa las ecuaciones en su representación algebraica, esto es plantear las ecuaciones tal y como se plantearon en (1.1)-(1.5), una vez hecho se definen las opciones de resolución del sistema y por último se invoca al método *solve()* para encontrar la solución

Tras construir la wavelet con el banco de filtro hallado se procede a cargar la imagen de prueba usando *pydicom* como se muestra en el ejemplo de código 3.3. De la imagen se extrae la matriz de valores de los píxeles y se le aplica la transformada de wavelet de dos dimensiones.

Notese algo muy importante, en las mamografías digitales predominan los valores 0, que representan el color negro en la escala de grises, al aplicar la transformada de wavelets en dos dimensiones usando *pywt*, en algún punto se calcula la convolución

entre el filtro y una secuencia de ceros en la imagen, lo cual va a resultar en un valor igual a cero en la matriz de detalle. Dado que se usa la cercanía a este valor como medida de similaridad entre el patrón y el fragmento de la muestra, podría representar un problema, llegando a producir un gran número de falsos positivos.

Para evitar este inconveniente se calcula la transformada de wavelet de forma manual, así, si en algún punto se calcula la convolución entre el vector  $\tilde{v}$  del filtro y una secuencia de al menos  $\frac{N^2}{4}$  ceros, no se considera el resultado obtenido como posible ocurrencia del patrón.

Al aplicar entonces la transformada de wavelet se obtienen las matrices  $cA$ ,  $cH$ ,  $cV$  y  $cD$ . De ellas se toma la última y se realiza la búsqueda de ceros que indiquen la ocurrencia del patrón. Suponga que se encuentra una posible ocurrencia del patrón en  $(i, j)$ , realizando el mismo análisis que en una dimensión, se tiene:

$$\begin{aligned} cD(i, j) = 0 &\Leftrightarrow D(img * \tilde{w}_{0,0})(i, j) = 0 \\ &\Leftrightarrow (img * \tilde{w}_{0,0})(2i, 2j) = 0 \\ &\Leftrightarrow \sum_{n=0}^{N-1} \sum_{m=0}^{N-1} \tilde{w}_{0,0}(n, m) img(2i - n, 2j - m) = 0 \end{aligned}$$

cuyo resultado indica la ocurrencia del patrón entre  $(2i - N + 1, 2j - N + 1)$  y  $(2i, 2j)$ .

Un punto muy importante que se tiene en cuenta es la ocurrencia del patrón en una imagen con ruido. En estos casos es casi imposible lograr una medida  $S$  cercana a la unidad, por esta razón se emplea un umbral de detección, de esta forma, cualquier valor en la medida  $S$  que sobrepase este umbral es considerada una ocurrencia del patrón, mientras que de quedar por debajo, se considera que el patrón no se localizó en la posición actual.

Para analizar el funcionamiento del algoritmo se mostrará un ejemplo con imágenes. Suponga que se desea reconocer la ocurrencia de una letra (digamos **c**, ver figura 3.9) en una lista de palabras:

Para este caso se toma  $N = 8$  como la dimensión de los vectores del filtro. Tras leer la imagen con `pydicom`, se obtiene la siguiente representación matricial:

$$m = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0,0015 & 0,0039 & 0,0039 & 0,0039 & 0,002 & 0 & 0 \\ 0 & 0,0015 & 0,0038 & 0,0008 & 0 & 0,0001 & 0,0026 & 0,0008 & 0 \\ 0,0001 & 0,0027 & 0,002 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0,0005 & 0,0036 & 0,0014 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0,0005 & 0,0036 & 0,0014 & 0 & 0 & 0 & 0 & 0 & 0,0001 \\ 0,0001 & 0,0034 & 0,002 & 0 & 0 & 0 & 0 & 0 & 0,0005 \\ 0 & 0,0015 & 0,0038 & 0,0008 & 0 & 0,0001 & 0,0026 & 0,0008 & 0,0001 \\ 0 & 0 & 0,0015 & 0,0039 & 0,0039 & 0,0039 & 0,002 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

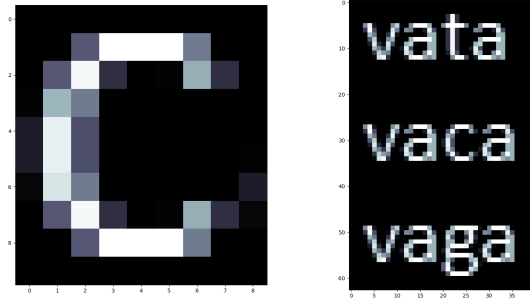


Figura 3.9: Ejemplo de un patrón y una imagen de prueba.

Con los valores de esta matriz se construyen las ecuaciones de *matching* y junto con el resto de las condiciones se forma el sistema de ecuaciones. A la hora de resolverlo se usa primeramente una aproximación inicial a la raíz proporcionado por el método de Runge Kutta. Los valores  $x_0$  son generados de forma aleatoria y por tanto la raíz encontrada en cada ejecución podría cambiar. Por ejemplo, en este caso, antes de aplicar Runge Kutta:

$$x_0 = (0,773, 0,7211, 0,382, 0,0948, 0,0963, 0,963, 0,2968, 0,1797)$$

mientras que, al aplicar Runge Kutta se tiene:

$$x_* = (-0,0607, -2,5645, 4,3419, 14,4576, -0,7601, -21,0409, -2,3254, -8,2964)$$

el valor de  $x_*$  es usado entonces como valor inicial al aplicar el método de gekko. Tras resolver el sistema nuevamente se obtiene:

$$v = (0,0061, -0,0121, -0,0813, -0,0222, 0,6232, -0,7612, 0,141, 0,0708)$$

A partir del valor de  $v$  obtenido se calculan el resto de los vectores que conformarán el banco de filtros:

$$u = (-0,0121, 0,0061, 0,0708, -0,141, -0,7612, -0,6232, -0,0222, 0,0813)$$

$$\tilde{v} = (0,0061, 0,0708, 0,141, -0,7612, 0,6232, -0,0222, -0,0813, -0,0121)$$

$$\tilde{u} = (-0,0121, 0,0813, -0,0222, -0,6232, -0,7612, -0,141, 0,0708, 0,0061)$$

con el cual se construye la Shepelet.

Para el paso de análisis de la imagen en busca de la ocurrencia del patrón, se carga el archivo de la misma a partir de su dirección tal como se muestra en el

Ejemplo de código 3.3. Una vez se tiene la forma matricial de la imagen de muestra, se determina la transformada de wavelet de dos dimensiones usando la Shapelet creada. La figura 3.10 muestra las imágenes obtenidas tras aplicar las transformadas junto con los valores de la medida  $\mathbb{S}$  para  $\alpha = 0,1$  aplicada a la matriz  $cD$ .

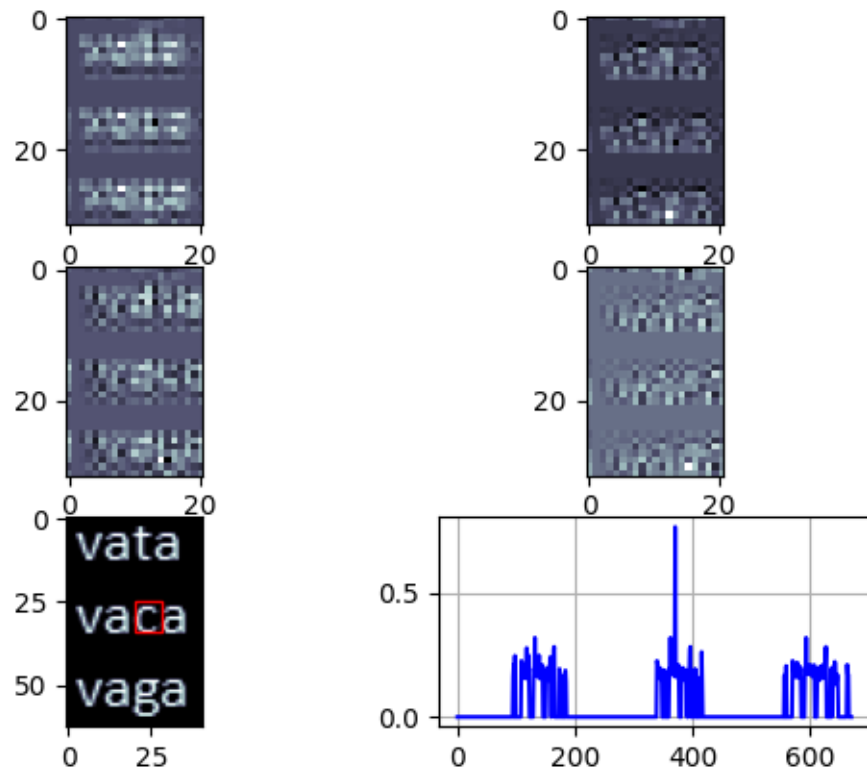


Figura 3.10: Imágenes de aproximación, detalles horizontales, detalles verticales, detalles (comúnmente llamados diagonales), imagen con el patrón señalado y medida  $\mathbb{S}$ , en ese orden.

Como se puede apreciar hay una amplia diferencia entre la medida  $\mathbb{S}$  del punto en el que coincide el patrón y el resto de los puntos. Este ejemplo muestra un buen comportamiento del algoritmo, veamos entonces que ocurriría en casos más complejos como es el de detectar tumores.

Para el siguiente ejemplo, se tomó una anomalía, se le redujo la dimensión hasta ser tratable por el algoritmo y posteriormente se determinó la Shapelet asociada

al patrón. Luego se tomó una mamografía digital con el tumor y se le aplicó la transformada usando la Shapelet antes creada. La figura 3.11 muestra los resultados obtenidos en el proceso de detección.

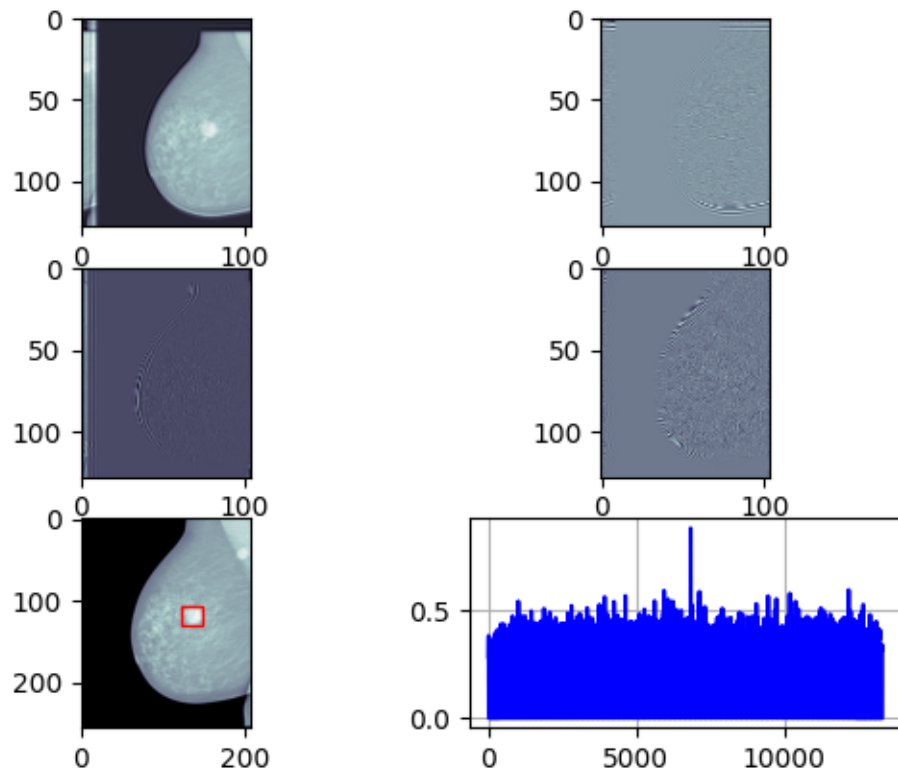


Figura 3.11: Resultados obtenidos en el proceso de detección de la anomalía en la mamografía.

Para poner a prueba el algoritmo, se tomó una colección de mamografías digitales tanto de personas sanas, como de personas con cáncer y con presencia de un posible tumor. Se tomaron varios patrones de anomalías y se extrajo un patrón de imagen que se repitiera en la mayor cantidad de anomalías, posteriormente se creó la Shapelet a partir de ese patrón y se ejecutó el algoritmo para cada imagen de muestra. La figura 3.12 muestra los valores de precisión y recobrado para un aumento de los casos de prueba.

La idea inicial de tomar un patrón común entre las anomalías permitió que se alcanzaran los valores mostrados de precisión y recobrado. El proceso de detectar un

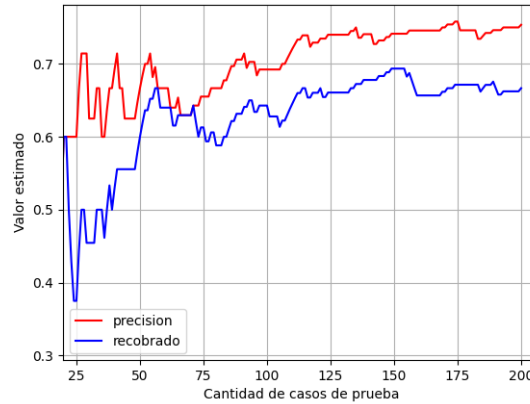


Figura 3.12: Valores de precisión y recobrado obtenidos durante el aumento de los datos procesados, de una colección de 200 mamografías.

tumor en una mamografía se ve afectado por varios factores: forma del patrón, tamaño, textura, opacidad, todos fácilmente clasificados como ruido en el proceso de detección. Si se usara un solo patrón serían muchos los falsos negativos que daría pues no sería capaz de reconocer otras anomalías con forma o textura distinta.

Este software fue creado basado en la teoría antes expuesta, por lo que solo se encuentra en su fase experimental. Mejorando varios aspectos como la resolución del sistema, la representación de las imágenes y la detección efectiva del patrón, podría convertirse en una muy buena herramienta para identificar anomalías y llegar incluso a salvar vidas con la detección temprana del cáncer de mama.

# Conclusiones

Se propuso un método para la detección de anomalías en mamografías digitales usando la transformada de Shapelet en dos dimensiones. Para ello se brindaron un conjunto de herramientas y resultados que pueden ser empleados en futuros estudios de la teoría wavelets y no solo para el reconocimiento en imágenes. El algoritmo propuesto aunque presenta un buen número de fallos a la hora de tratar con imágenes que contienen el patrón con ruido, razón por la cual se usó la alternativa de seleccionar una porción de la anomalía para los experimentos finales, posee una gran efectividad a la hora de reconocer patrones bien definidos como fue el caso de detectar una letra en un conjunto de palabras, resultado que indica que, aunque quedan muchos aspectos por mejorar el uso de shapelets para el reconocimiento de patrones, es un camino correcto.

Cabe destacar también los resultados obtenidos durante la experimentación en una dimensión y el análisis de los métodos de resolución de sistemas de ecuaciones, pues permitieron sentar las bases para el desarrollo del concepto de Shapelet bidimensional.

De manera general, se cumplió el objetivo principal que era construir una base wavelet que permitiera la detección de anomalías en mamografías y ,se reitera, aunque quedan varios aspectos que mejorar, se considera que este método pueda ser utilizado en un futuro en la medicina para detectar el cáncer de mama en una etapa temprana.



# Recomendaciones

Repasando los puntos que pueden mejorarse del algoritmo propuesto, se encuentra encontrar una forma eficiente y efectiva de resolver el sistema de ecuaciones que permita considerar patrones con dimensiones mucho mayores a la actual. Otro punto es la sensibilidad al ruido que presenta el algoritmo, pues usar solo una porción de la anomalía puede desembocar en un gran número falsos positivos.

Un aspecto que no se mencionó pero que cabe puntualizar para futuras investigaciones sobre el tema es la posibilidad de usar patrones que aparecen en distinta proporción en la imagen a analizar. En la presente investigación las imágenes que se usaron de experimento eran imágenes de tamaño real reducidas con la DWT un número de veces igual al del patrón, por lo que siempre se encontraban en la misma proporción en los casos en los que aparecía. Es necesario tener en cuenta que un patrón en distinta proporción significa información extra o faltante, por lo que el proceso de detección podría tornarse más complejo.

# Bibliografía

- Ballard D., S. J. (1973). Tumor detection in radiographs. Computers and Biomedical Research (vid. pág. 3).
- Chan, H., Doi, K., Galhotra, S., Vyborny, C. J., MacMahon, H. & Jokich, P. M. (1987). Image feature analysis and computer-aided diagnosis in digital radiography. I. Automated detection of microcalcifications in mammography (vid. pág. 4).
- de la Salud (OMS), O. M. (s.f.). Cáncer. (Vid. pág. 1).
- de la Salud (OPS), O. P. (s.f.). Cáncer de mama. (Vid. pág. 1).
- Frazier, M. W. (1999a). *An Introduction to Wavelets Through Linear Algebra*. (Vid. pág. 7).
- Frazier, M. W. (1999b). *An Introduction to Wavelets Through Linear Algebra*. (Vid. pág. 8).
- Gargour, C., Gabrea, M., Ramachandran, V. & Lina, J.-M. (2009). A Short Introduction to Wavelets and Their Applications (vid. pág. 5).
- Guido, R. C. (2017). Fusing time, frequency and shape-related information: Introduction to the Discrete Shapelet Transform's second generation (DST-II) (vid. pág. 9).
- Guido, R. C., Junior, S. B., Vieira, L. S., Sanchez, F. L., Maciel, C. D., Pereira, J. C., Scalassara, P. R. & Fonseca, E. S. (2008). Introduction to the Discrete Shapelet Transform and a new paradigm: joint time-frequency-shape analysis (vid. pág. 9).
- José Salvado, B. R. (2005). Detection of Calcifications in Digital Mammograms using Wavelet Analysis and Contrast Enhancement (vid. pág. 4).
- Logan Beal, J. H. (s.f.). GEKKO Optimization Suite. (Vid. pág. 27).
- Masahiko Yachida, M. I. & Tsuji, S. (1980). A Plan-Guided Analysis of Cineangiograms for Measurement of Dynamic Behavior of Heart Wall (vid. pág. 3).
- Mecías, A. L. (2016). Notas de clases, Matemática Numérica, Facultad de Matemática y Computación de La Universidad de La Habana (vid. pág. 5).
- Pizer, S. M. & Todd-Pokropek, A. E. (1978). Improvement of Scintigrams by Computer Processing (vid. pág. 3).
- Pydicom. (s.f.). (Vid. pág. 32).
- Wavelets Documentation. (s.f.). (Vid. pág. 29).

Zhi Shi, B.-Y. P., Hai-Yan Xing. (2009). TWO-DIMENSIONAL BIORTHONORMAL  
WAVELETS ON  $\mathbb{Z}N_1 \times \mathbb{Z}N_2$  (vid. págs. 14, 15, 17).