

Medición del ritmo cardíaco por medios ópticos

MÉTODOS NUMÉRICOS AVANZADOS

INSTITUTO TECNOLÓGICO DE BUENOS AIRES

BALAGUER, Pedro
55795

BENÍTEZ, Julián
56283

GARRIGÓ, Mariano
54393

PERAZZO, Matías
55024

SAQUÉS, M. Alejo
56047

Resumen

Palabras clave:

1. Transformada de Fourier

1.1. Series de Fourier

Una serie de Fourier permite descomponer cualquier función periódica en una suma, finita o no, de senos y cosenos.

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos(nx) + \sum_{n=1}^{\infty} b_n \sin(nx) \quad (1)$$

Donde a_n y b_n son los **coeficientes de Fourier** de la serie de Fourier de la función $f(t)$.

Con el uso de propiedades trigonométricas, se puede llegar a una expresión reducida:

$$f(x) = \frac{A_0}{2} + \sum_{n=1}^N A_n \cdot \sin\left(\frac{2\pi nx}{P} + \phi_n\right) \quad (2)$$

$$A_n = \sqrt{a_n^2 + b_n^2} \quad (3)$$

$$\phi_n = \arctan_2(a_n, b_n) \quad (4)$$

Con $N \geq 1$, siendo $f(x)$ integrable en un intervalo $[x_0, x_0 + P]$. Se puede notar que A_n representa el peso del n -ésimo seno dentro de la función periódica $f(x)$.

1.2. Transformada continua de Fourier

La transformada continua de Fourier (**TCF**) puede ser entendida como una extensión de las series de Fourier, pero donde el período de la función puede tender a infinito. La transformada de Fourier permite determinar cuál es el *grado de presencia* de cierta frecuencia en una función o señal, es decir, permite transformar un dominio temporal a uno de frecuencias. Nótese que esto no se limita a funciones variables en el tiempo, pero con objetivo de unificar el vocabulario se le llama dominio temporal al dominio de la función original. La transformada de Fourier se define de la siguiente manera:

$$\hat{f}(\xi) = \int_{-\infty}^{\infty} f(x) \cdot \exp^{-2\pi i x \xi} dx \quad (1)$$

Con x perteneciendo al dominio de los tiempos, y ξ al de las frecuencias.

Utilizando la fórmula de Euler, se ve que la transformada de Fourier puede ser expresada con senos y cosenos:

$$\exp^{ix} = \cos x + i \sin x$$

De forma similar a las series de Fourier, donde el valor A_n representaba el peso del n -ésimo seno, la transformada de Fourier permite encontrar cada uno de los componentes sinusoidales con frecuencia ξ y su respectivo peso, permitiendo *reconstruir* la función original.

A continuación sigue un ejemplo ilustrativo del proceso.

En primer lugar, se considera una función $f(t)$, es decir, con dominio temporal $t \in [0, 2,5]$:

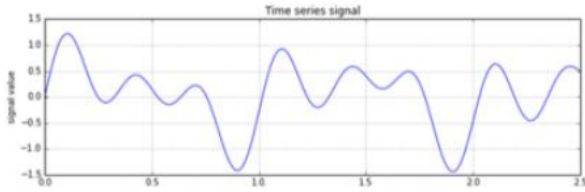


Figura 1: Alguna función con dominio temporal

Luego, cada punto (o *vector*), puede proyectarse en un sistema de coordenadas con los ejes *Reales* e *Imaginarios* con una frecuencia ξ dada:

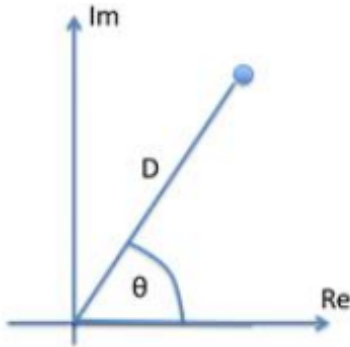


Figura 2: Ejes coordenados

Con $\Phi = 2\pi t\xi$ y $D = f(t)$

Proyectando cada punto de f para todo t en el dominio de la muestra, fijando distintas frecuencias ξ , obtendríamos, para este ejemplo, los siguientes resultados:

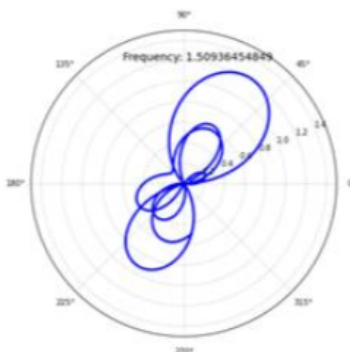


Figura 3: Ejemplo para $\xi = 1,509$

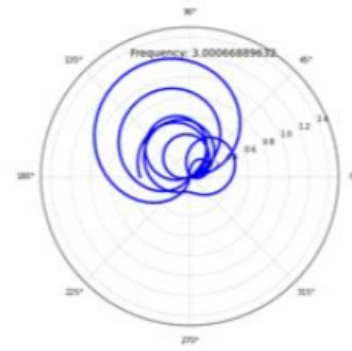


Figura 4: Ejemplo para $\xi = 3,0007$

Las figuras 3 y 4 corresponden a la siguiente expresión para alguna frecuencia ξ dada:

$$f(t) \exp(-2i\pi t\xi) \quad (2)$$

Finalmente, para obtener el peso de la función sinusoidal con frecuencia ξ , se suman los módulos de los vectores de la proyección. Luego, se obtiene una función $\hat{f}(\xi)$ que representa el peso de cada frecuencia ξ en la función $f(t)$ dependiente del tiempo.

1.3. Transformada discreta de Fourier

La transformada discreta de Fourier (**TDF**) guarda la misma esencia que la continua. Esta transformada permite representar el dominio de las frecuencias de una señal a partir del muestreo de una señal continua cada intervalos de tiempo constantes. La expresión de la **TDF** es la siguiente:

$$X_k = \sum_{n=0}^{N-1} x_n \cdot \exp^{-\frac{2\pi i}{N} kn} \quad (1)$$

Lo cual transforma una secuencia de N números complejos x_0, x_1, \dots, x_{N-1} , en otra secuencia de sobre el mismo cuerpo X_0, X_1, \dots, X_{N-1} . Aquí $0 \leq k \leq N-1$. La interpretación es que $\hat{f}(\xi_k) = X_k$.

Se puede notar que la expresión de la **TDF** es muy similar a la de la **TCF**, con la única diferencia de que la primera utiliza una sumatoria en vez de una integral, ya que en este caso se está tratando con una cantidad discreta de valores. Por ende, salvando esta diferencia, la interpretación gráfica del proceso es análogo al caso anterior.

1.3.1. Algoritmo por fuerza bruta

El algoritmo por fuerza bruta es, en esencia, lo que su nombre indica: utilizar la expresión de la **TDF** exhibida más arriba para realizar el cálculo de la transformada. Esto implica que, por cada X_k , se realiza una sumatoria sobre N productos, por lo que el orden algorítmico de esta implementación es $O(N^2)$.

1.3.2. Fast Fourier Transform

Una versión mejorada de la implementación de la **FFT**, propuesta por Cooley y Tukey (1965), permite reducir el orden computacional a $O(n \cdot \log n)$. La idea se basa en el aprovechamiento de las simetrías de $\exp^{-\frac{2\pi i}{N}}$ en la expresión de la **TDF**.

Sea $W_n = \exp^{-\frac{2\pi i}{N}}$. Luego, puede verse que:

$$W_n^{kn} = \exp^{-2\pi i k} = 1 \quad (2)$$

Luego, se tienen las siguientes propiedades:

Propiedad 1 (Simetría conjugada compleja)

$$W_n^{k(N-n)} = W_n^{-kn} = (W_n^{kn})^*$$

Propiedad 2 (Periodicidad en n y k)

$$\begin{aligned} W_n^{kn} &= W_n^{k(N+n)} \\ &= W_n^{(k+N)n} \end{aligned}$$

La transformada rápida de Fourier se basa en el concepto de *divide and conquer*: asumiendo la cantidad de entradas $N = 2^m$ se subdivide el problema en dos de tamaño $\frac{N}{2}$ a partir de los cuales es posible calcular el primero. El enfoque más común es dividir el cálculo de la transformada de Fourier según índices pares e impares:

$$X_k = \sum_{n=0}^{N-1} x_n W_N^{kn} \quad (3)$$

$$= \sum_{n \text{ par}} x_n W_N^{kn} + \sum_{n \text{ impar}} x_n W_N^{kn} \quad (4)$$

$$= \sum_{r=0}^{\frac{N}{2}-1} x_{2r} W_N^{k2r} + \sum_{r=0}^{\frac{N}{2}-1} x_{2r+1} W_N^{k(2r+1)} \quad (5)$$

$$= \sum_{r=0}^{\frac{N}{2}-1} x_{2r} (W_N^2)^{kr} + W_N^k \sum_{r=0}^{\frac{N}{2}-1} x_{2r+1} (W_N^2)^{kr} \quad (6)$$

$$\text{Como } W_N^2 = \exp^{-\frac{2\pi i}{N}} = \exp^{-\frac{2\pi i}{\frac{N}{2}}} = W_{\frac{N}{2}}.$$

Sea X_{p_k} la **TDF** para las muestras pares y X_{i_k} para las impares. Esto implica que se pueden calcular de forma independiente X_{p_k} y X_{i_k} , ambas con $\frac{N}{2}$ muestras en vez de N , luego combinando los resultados para obtener X_k .

Nótese que, gracias a la Propiedad 2, $X_{p_{k+\frac{N}{2}}} = X_{p_k}$ (ídem para X_i) Luego:

$$X_k = \begin{cases} X_{p_k} + W_N^k X_{i_k}, & \text{si } 0 \leq k < \frac{N}{2} \\ X_{p_{k-\frac{N}{2}}} + W_N^k X_{i_{k-\frac{N}{2}}}, & \text{si } \frac{N}{2} \leq k < N \end{cases} \quad (7)$$

También se puede obtener:

$$\begin{aligned} X_k &= X_{p_k} + W_N^k X_{i_k} \\ X_{k+\frac{N}{2}} &= X_{p_k} - W_N^k X_{i_k} \end{aligned}$$

A partir de estos resultados se puede realizar un algoritmo recursivo para calcular la **TDF** con orden computacional $O(n \cdot \log n)$. A continuación, presentamos el pseudo-código de la transformada rápida de Fourier siguiendo este método:

```
def FFT(X):
    N = len(X)
    if N > 1:
        even = X[::2]
        odd = X[1::2]
        FFT(even)
        FFT(odd)
        for k in range(N/2):
            e = X[k]
            o = X[k+N/2]
            w = exp(-2*pi*1j/n)
            X[k] = e + w * o
            X[k+N/2] = e - w * o
    return X
```

2. Método

TODO AFTER LUNCH BREAK

3. Resultados

A continuación, se exhibirán los tiempos de ejecución entre las diferentes implementaciones del al-

goritmo *fft* realizadas, como así también de los resultados obtenidos a la hora de calcular el ritmo cardíaco de un individuo.

3.1. Algoritmos *fft*

A continuación, se presenta una tabla comparativa de tiempos de ejecución entre la implementación recursiva e iterativa del algoritmo *fft Cooley-Tukey*.

N	Recursivo	Iterativo	Mejora
512	.009	.003	x3
1024	.02	.006	x3.34
2048	.046	.014	x3.29
4096	.096	.029	x3.21
8192	.201	.06	x3.35

Cuadro 1: Comparación entre implementaciones de *Cooley-Tukey*

Como podrá verse en la Tabla 1, la implementación iterativa del algoritmo *Cooley-Tukey* es claramente del mismo orden algorítmico, pero en torno a 3,3 veces más rápido. Se arguye que la mejora en la *performance* proviene de la eliminación del *overhead* generado por los *stackframes* producidos por la implementación recursiva en cada llamada.

3.2. Mediciones del ritmo cardíaco

Se han tomado 5 muestras de un individuo en diferentes partes del cuerpo, variando el uso del **LED** del dispositivo:

1. Índice izquierdo (cubriendo el **LED** con el mismo),
2. Pulgar derecho (sin cubrir el **LED**),
3. Antebrazo,
4. Índice derecho (sin cubrir el **LED**),
5. Índice derecho (con el **LED** apagado).

Al momento de tomar las capturas, el individuo se encontraba en reposo. Se procuró que el mismo realizara inhalaciones y exhalaciones a intervalos regulares de aproximadamente 2 segundos, instruyéndole que realizara las mismas de manera calma.

A modo de control, se tomó una serie de muestras del ritmo cardíaco del individuo con mecanismos tradicionales. Para mayor precisión, se utilizó un estetoscopio y se contó durante el lapso de 1 minuto la cantidad de pulsaciones. Los siguientes parámetros describen la muestra:

- $N = 14$
- $\bar{X} = 78,929$
- $\sigma = 4,0089$

A continuación se exhibirán los resultados utilizando diferentes métodos para obtener un valor escalar que represente el *brillo* de la imagen para un instante dado, tomando una trama en escala de grises.

Debe notarse que, en el análisis a continuación, se asume que el ritmo cardíaco real permanece constante entre cada una de las muestras tomadas. Dadas las circunstancias en las que se re han realizado las mediciones, esta asunción puede tener un grado alto de validez. Sin embargo, eventuales variaciones podrían influir sobre la precisión de los errores presentados a continuación.

3.3. Región cuadrada con vértice en el centro

Para este caso, se ha tomado una región cuadrada de 30X30 con vértice en el centro de la imagen. Este caso es el utilizado por la Cátedra en el código de ejemplo.

Muestra	Ppm.	Error
1	73.815	6.93 %
2	75.592	4.41 %
3	86.150	9.15 %
4	80.854	2.44 %
5	84.316	6.83 %

Cuadro 2: Región 30X30 con vértice en el centro

Como podrá verse en el Cuadro 2, la aproximación con menor error relativo al promedio ha sido la de la muestra correspondiendo al dedo índice sin cubrir el **LED**. Por otro lado, la que mayor error relativo parecería mostrar es el caso del antebrazo.

3.4. Promedio de toda la imagen

En este caso, se han promediado todos los puntos de la imagen en escala de grises.

Muestra	Ppm.	$ Error $
1	73.815	6.93 %
2	86.139	9.13 %
3	87.908	11.38 %
4	80.853	2.44 %
5	59.724	32.16 %

Cuadro 3: Promedio de toda la imagen

Utilizando el método anterior, el error relativo en la muestra 5 era comparable al de las otras muestras. En el Cuadro 3, podrá verse que dicho error, con este método, excede con creces el de las otras muestras. Esto indicaría que utilizar el **LED** podría ser un requisito indispensable a la hora de aproximar el ritmo cardíaco.

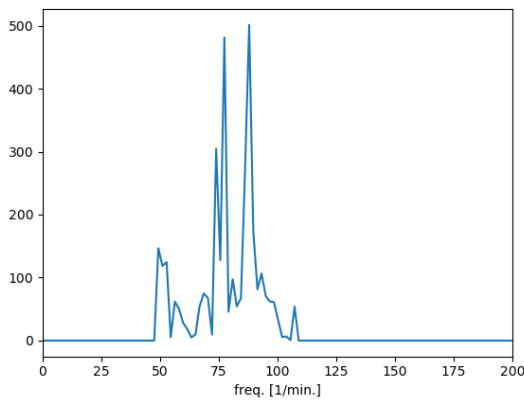


Figura 5: Muestra 3: proximidad entre picos

Otro caso cuyo error creció con respecto al obtenido en el método anterior es el de la muestra 3. Sin embargo, como se podrá ver en la Figura 5, existen dos picos de magnitud comparable claramente distinguibles uno del otro. El mayor es, razonablemente, el que corresponde a la frecuencia 87,908. El que le sigue, corresponde al valor de 77,016, lo que, de tomarse como valor del ritmo cardíaco, marcaría un $|Error| = 2,48 \%$. Esto podría ser absolutamente azaroso, pero la eventual precisión del segundo valor en magnitud suscita curiosidad sobre la perspectiva de poder tomar la frecuencia del segundo mayor pico al tomar la medición en el antebrazo.

3.5. Región cuadrada con vértice en el centro, interpolando puntos

Para este último caso, se ha tomado una región cuadrada de 200×200 con vértice en el centro de la imagen, interpolando de a 5 píxeles. El objetivo de esto es maximizar la superficie cubierta, a la vez que se toma una cantidad similar de píxeles que en el primer caso. Dado que se está filmando con la lente inmediatamente sobre la piel, la distancia real entre los píxeles es ínfima. Luego, al interpolar muy probablemente se obtenga una aproximación cercana al verdadero promedio de todos los píxeles de la región.

Muestra	Ppm.	$ Error $
1	73.815	6.93 %
2	73.834	6.90 %
3	87.908	11.38 %
4	80.853	2.44 %
5	59.724	32.16 %

Cuadro 4: Interpolando

Como podrá verse en la Tabla 4, salvo una aparente mejoría en la muestra 2, los valores del $|Error|$ son aproximadamente similares a los del caso anterior.

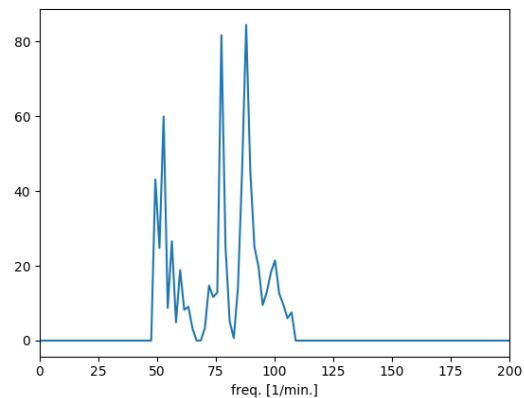


Figura 6: Muestra 3: proximidad entre picos

Tal como se ha observado en el método de promediado anterior, en la muestra 3, tras una inspección del gráfico de frecuencias, se observan dos picos de magnitud comparable. En este caso, el valor de la segunda frecuencia más representativa es de 77,419, lo que representaría un $|Error| = 1,95 \%$,

el menor de todos los errores relativos obtenidos hasta el momento.

3.6. Nueva muestra en el antebrazo

En los apartados anteriores, se ha notado una característica particular en la gráfica de frecuencias para los vídeos tomados sobre el antebrazo. Se ha considerado oportuno indagar sobre dicho punto, tomando 2 nuevas mediciones sobre dicha parte del cuerpo del mismo sujeto de pruebas. A continuación se exhiben las gráficas de frecuencias:

ruido al tomar la filmación. Además, dada la falta de precisión en todas las medidas en las que se utilizó el antebrazo, podría decirse que dicho sector del cuerpo no es apropiada para estimaciones certeras.

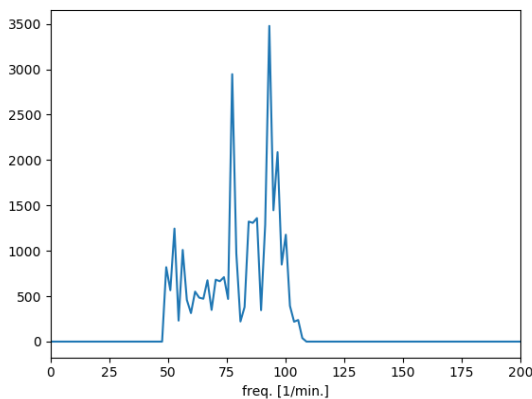


Figura 7: Muestra 6: proximidad entre picos

Como podrá notarse en la Figura 7, el caso parecería ser idéntico al de la muestra 3: el pico mayor se encuentra en la frecuencia 93,079, y el que le sigue, en 77,4194.

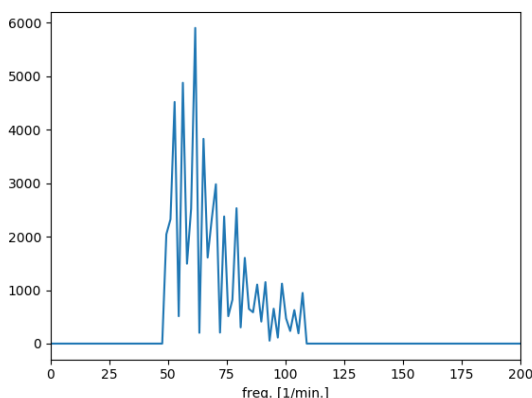


Figura 8: Muestra 6: proximidad entre picos

Sin embargo, en la Figura 8, la estimación obtenida es de 61,535, y el fenómeno que se observó antes es inexistente. Luego, se podría concluir que lo antes observado responde al azar producto del