



UNIVERSIDAD DE BUENOS AIRES
FACULTAD DE INGENIERÍA

2° CUATRIMESTRE DE 2017

86.05

SEÑALES Y SISTEMAS

Trabajo Práctico n°1
Detección de latidos cardíacos

Integrante:
Garcias, Ezequiel <garciaezequiel91@gmail.com>

Padrón:
93191

27 de noviembre de 2017

Índice

1. Introducción	2
2. Objetivos	2
3. Enunciado	3
4. Desarrollo	5
4.1. Ejercicio 1.	5
4.2. Ejercicio 2.	6
4.3. Ejercicio 3.	7
4.4. Ejercicio 4.	8
4.5. Ejercicio 5.	10
4.6. Ejercicio 6.	12
4.7. Ejercicio 7.	13
4.8. Ejercicio 8.	14
4.9. Ejercicio 9.	15
4.10. Ejercicio 10.	16
4.11. Ejercicio 11.	22
4.12. Ejercicio 12.	23
4.13. Ejercicio 13.	24
5. Conclusiones	27

1. Introducción

Hoy en día es muy común realizar un estudio para poder ver los latidos del corazón por medio de instrumentación médica, la cual nos brinda información acerca de nuestro estado de salud.

Sin embargo debido a la tecnología, es posible para leer señales de la vida cotidiana. La mayoría de los celulares que las personas poseen tienen una gran capacidad de procesamiento.

De modo que utilizando un algoritmo nos permite saber las pulsaciones por minuto de un corazón, por ende ya no es necesario realizar un control médico para conocer nuestro ritmo cardíaco.

2. Objetivos

El objetivo del presente trabajo es, en primer lugar, realizar un análisis en tiempo y frecuencia de la señal FPG obtenida a partir de la grabación de video de un smartph-
hone. Posteriormente, se implementará una serie de filtros para obtener la componente *AC* de la señal, la cual se utilizará finalmente durante la implementación del detector de latidos. En lugar de utilizar el video directamente como señal, lo cual resulta de-
mandante con la memoria, se trabajará con las intensidades de color para cada frame del video en cuestión.

3. Enunciado

Detección de latidos cardíacos.

1. Cargar el archivo *intensidad_RGB.mat* y visualizar la señal RGB en superposición (3 canales). Elija alguno de los 3 canales que, según su criterio, posea la señal más útil a los efectos de analizar la dinámica de los latidos y utilícelo para realizar los puntos siguientes.
2. Del gráfico del punto anterior, estimar aproximadamente los latidos por minuto (LPM). Identifique el momento a partir del cual la frecuencia cardiaca comienza a incrementarse, y en cuánto se incrementa.
3. Rehacer el punto anterior, pero utilizando DFT. ¿Puede en este caso identificar cuándo se da el cambio de LPM?
4. Diseñar un filtro pasa-banda tipo Butterworth con banda de paso entre 0.5 Hz y 10 Hz. Graficar respuesta en frecuencia (módulo y fase), diagrama de polos y ceros y respuesta al impulso (sugerencia: usar las funciones *butter* y *fvtool* de Matlab). ¿Qué papel juega el orden del filtro seleccionado en su diseño?
5. Filtrar la señal FPG utilizando el filtro diseñado en el punto anterior mediante la función *filter*. Grafique en superposición la señal original con la filtrada y comente acerca de:
 - a. Remoción de derivas
 - b. Cambios en la forma de la señal
 - c. Retardo de la señal filtrada respecto de la original
6. A partir de la respuesta en fase del filtro, calcule su retardo temporal y compare con lo observado en el punto 5c.
7. Implementar un filtrado IIR ida y vuelta para anular la fase del filtro (puede utilizar la función *filtfilt* de Matlab). Justificar teóricamente el funcionamiento de este tipo de filtrado y cuál resulta su ventaja. Filtrar nuevamente la señal FPG y comparar el resultado con lo obtenido en el punto anterior, particularmente en la forma de la señal y su retardo.
8. Realizar un espectrograma de la señal antes y después de filtrar, mediante la función *spectrogram* de Matlab (sugerencia: utilice la función *caxis* para saturar los colores del espectrograma y lograr una mejor visualización). Justificar la longitud de ventana elegida y comente acerca del resultado obtenido, relacionándolo con los puntos 2 y 3. Calcule la resolución en frecuencia de la ventana mediante DFT en Matlab. ¿Cómo haría para obtener mejor resolución en frecuencia y qué se pierde con esto?
9. Identificar en el espectrograma la zona donde el pulso se acelera. Observar con detenimiento los componentes de frecuencia que posee la señal y justificar el origen de cada uno (para esto último, necesitará hacer uso de la señal *audio_det.mat* para explicar todos los componentes observados).

10. Realizar un detector automático de latidos. El mismo debe tomar como entrada la señal FPG y producir como salida un vector de tiempos, donde cada tiempo corresponde a la detección de un latido en la señal. Para esto, se sugiere implementar los siguientes pasos:

- Filtrado pasa-banda de la señal, utilizando el filtrado del ejercicio 7.
- Filtro de derivada, implementado con un filtro FIR $h(n)=[-2 \ -1 \ 0 \ 1 \ 2]$.
- Normalización con energía instantánea: primero calcular la energía instantánea de la señal mediante un filtro MA1 de la señal del punto 10a elevada al cuadrado; luego dividir la señal del punto b por el vector obtenido. Esto tiene como objeto reducir el impacto de la presión sanguínea sobre el nivel de señal.
- Sobre-muestreo en un factor 4 para obtener mayor resolución temporal: implemente el sobre-muestreo utilizando la función `upsample` y diseñe un filtro interpolador FIR utilizando la herramienta `fdatool` de Matlab. Grafique: respuesta en frecuencia del filtro en módulo y fase, y señal original y sobre-muestreada en superposición.
- Detector de picos mediante umbral (puede definir como umbral un valor arbitrario).
- Gráfico en superposición de la señal con las marcas de los picos detectados. Sugieren-
cia para el punto e: en general, un pico en la señal producirá la detección de múltiples muestras por encima del umbral. Para reducirlas a sólo una, puede utilizar la función `diff` para evaluar la primera derivada sobre el vector de muestras detectadas (la cual contiene sólo unos y ceros), quedándose solo con las muestras positivas que resulten y buscando, para cada una de estas, la posición del máximo inmediato de la señal original dentro de un intervalo pre-establecido (este deberá contener mínimamente a la duración aproximada de los picos).

11. En base a los resultados del punto anterior, calcule y grafique el intervalo temporal instantáneo entre latidos (IBI: inter-beat interval) y los LPM instantáneos.

12. Opcional. Mejorar el detector de latidos aplicando las reglas de [5]:

- Establecer como regla que, si dos latidos se detectaron con una separación temporal menor a 200ms, sobrevive sólo aquel que corresponda al pico de la señal mayor entre ambos.
- Establecer como regla que si el IBI instantáneo aumenta repentinamente en al menos 1.5 veces entre muestras consecutivas, puede haberse perdido la detección de un latido. Cuando este sea el caso, realice una nueva detección de picos dentro del intervalo correspondiente, utilizando un umbral la mitad del nominal. Si de esta manera se halla un nuevo pico, distanciado al menos 360ms de la detección precedente, entonces clasificarlo como latido.

13. Opcional. Obtener su propia señal FPG mediante la cámara de su celular y aplicar los análisis y algoritmos desarrollados a esta señal. Para esto, realice una grabación tal como se explicó en la sección 1 y utilice los scripts auxiliares de Matlab para extraer los archivos de intensidades y audio del video. Nota: para un mejor resultado en la señal registrada, utilizar el dedo índice aplicando una leve presión sobre la lente, evitando resultar excesivamente suave o fuerte.

4. Desarrollo

4.1. Ejercicio 1.

Como podemos ver tenemos el gráfico de las tres señales superpuestas. Tenemos un total de 2636 muestras con una frecuencia de muestreo de 29.37Hz. Dando como resultado 90 segundos de video.

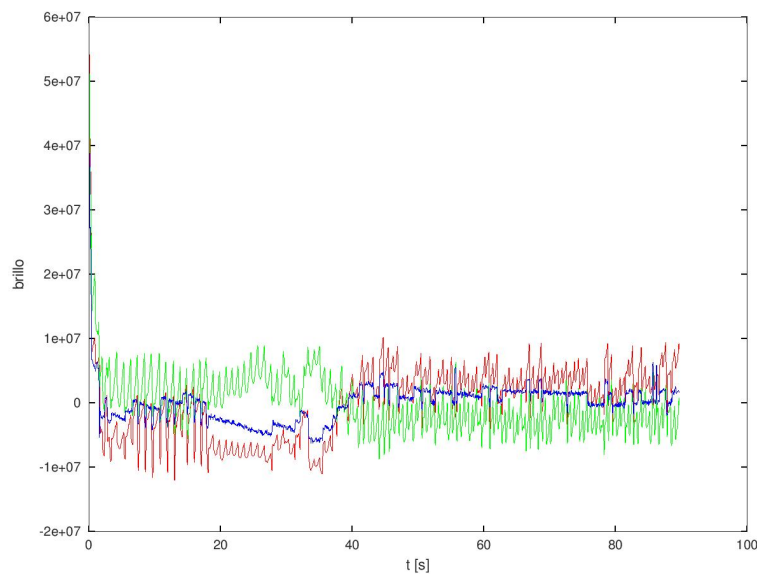


Figura 4.1 – Superposición tres señales, Roja Verde y Azul.

La señal azul no presenta la altura de los picos suficientemente marcados para lograr obtener información útil de ella. Las señal Roja y Verde pueden ser consideradas para obtener el resultado esperado. Finalmente opte por trabajar con la señal verde.

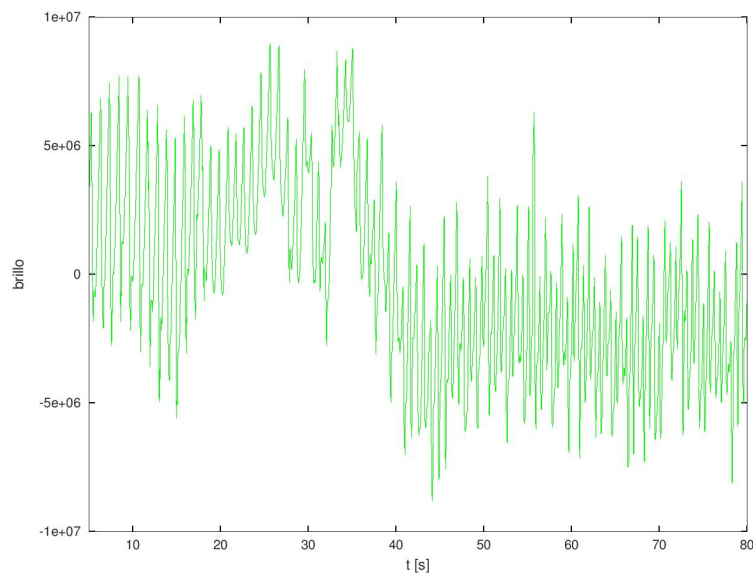


Figura 4.2 – Señal Verde.

4.2. Ejercicio 2.

Para poder observar la variación de latidos opte por graficar la señal verde cada 20 segundos. Finalmente al contar manualmente notamos como cada 20 segundos se observa una aceleración en el ritmo cardíaco.

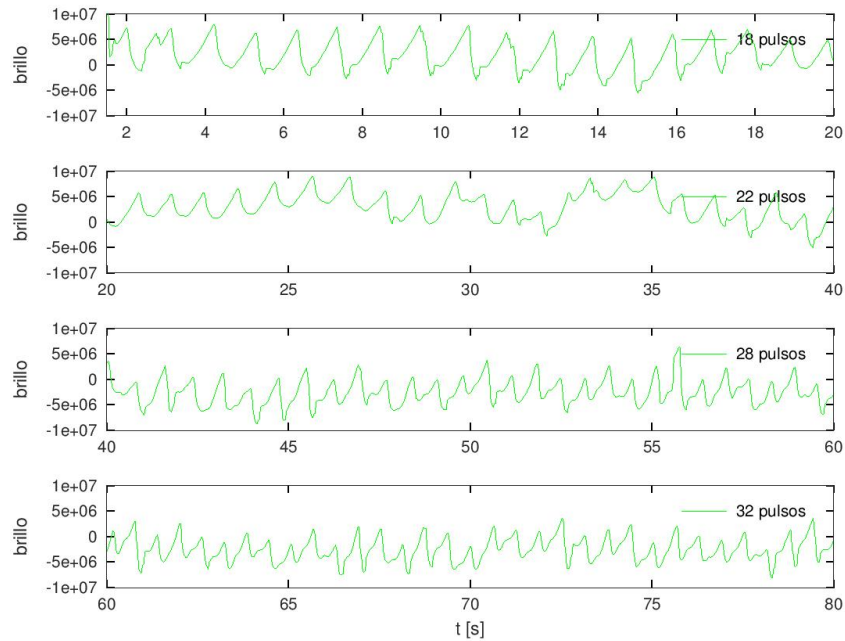


Figura 4.3 – Señal Verde conteo de pulsos.

Dando como resultado.

Intervalo [s]	LPM [latidos/min]
1.5 - 20	58
20 - 40	66
40 - 60	84
60 - 80	96

Se puede ver claramente como la frecuencia cardiaca crece a partir del segundo 40.

4.3. Ejercicio 3.

Luego de realizar una DFT a la señal verde, se nos pide hallar información acerca de los latidos por minuto.

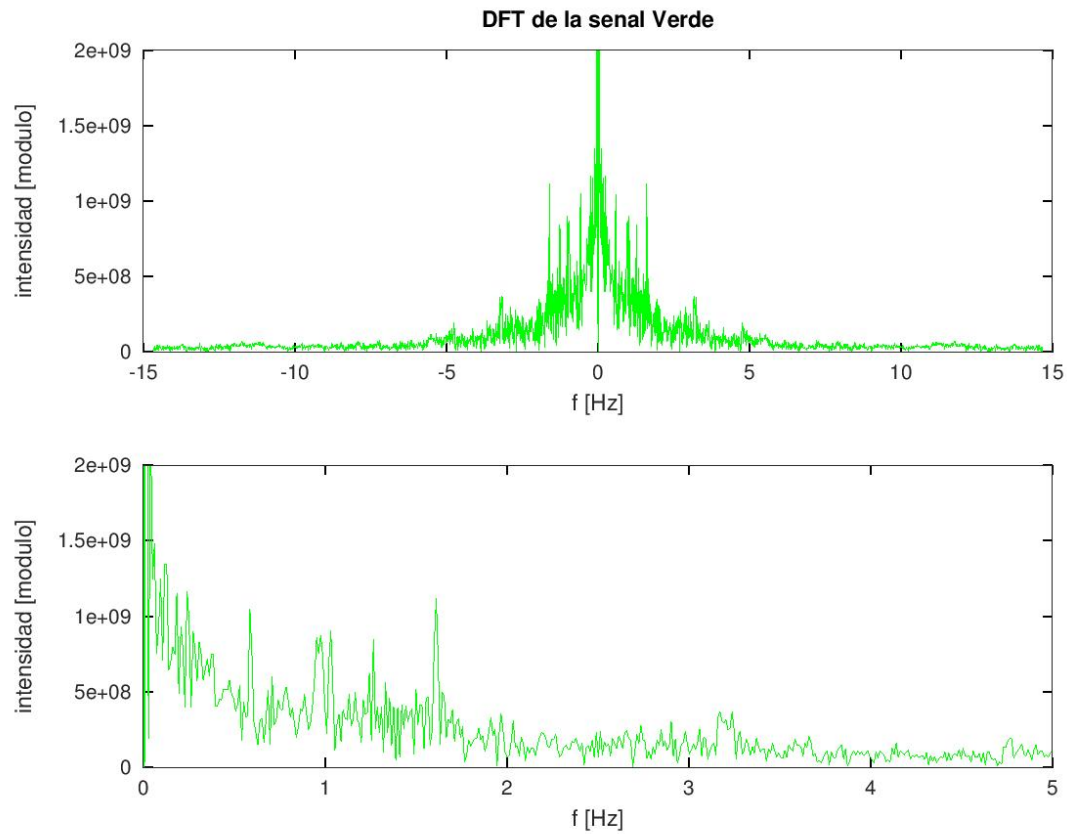


Figura 4.4 – Señal Verde Transformada Rápida de Fourier.

Como podemos observar la información aportada por la DFT solo nos dice cuales son las frecuencias presentes, pero no podemos obtener información sobre el cambio de ritmo cardíaco.

4.4. Ejercicio 4.

Para mejorar la lectura de la señal verde se propone diseñar un filtro pasabanda. El filtro tiene el fin de quitar las componentes de continua, bajas frecuencias y muy altas frecuencias que puede generar una lectura errónea de la información.

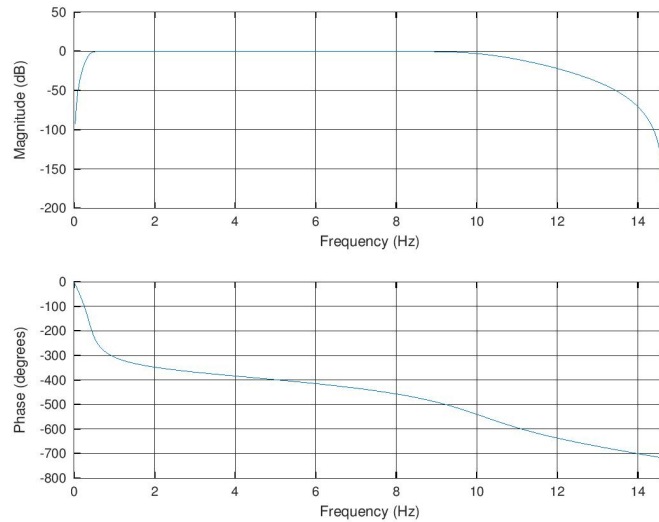


Figura 4.5 – Diagrama de filtro pasa banda.

Se utilizo un filtro de orden de 4 para el diseño, el orden del filtro genera cambios más abruptos para las frecuencias de corte seleccionadas.

Cada vez que se aumenta el orden, se generan polos cerca al cero para tener el cambio abrupto. En este caso en particular cada cero fue rodeado por 4 polos, dentro del circulo unitario para que se la ROC incluya el mismo y de esa forma se trate de una transferencia estable.

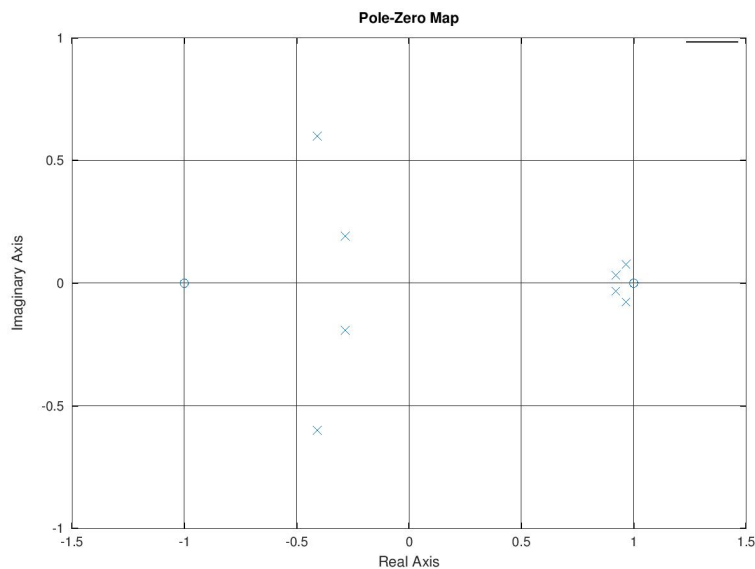


Figura 4.6 – Diagrama de polos y ceros.

Se utilizó la función **Filter** y se modeló una señal de entrada en forma de impulso para obtener la siguiente salida.

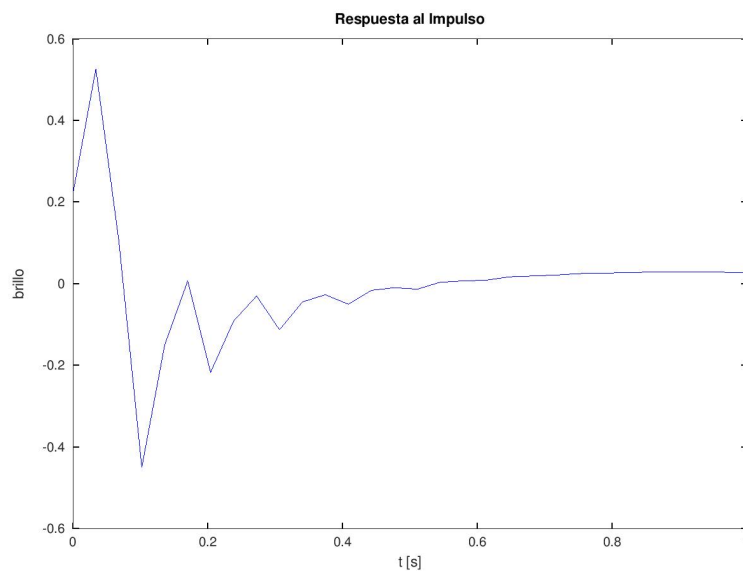


Figura 4.7 – Respuesta al impulso.

4.5. Ejercicio 5.

Se aplica el filtro diseñado anteriormente a la señal original.
Dando como resultado el siguiente gráfico:

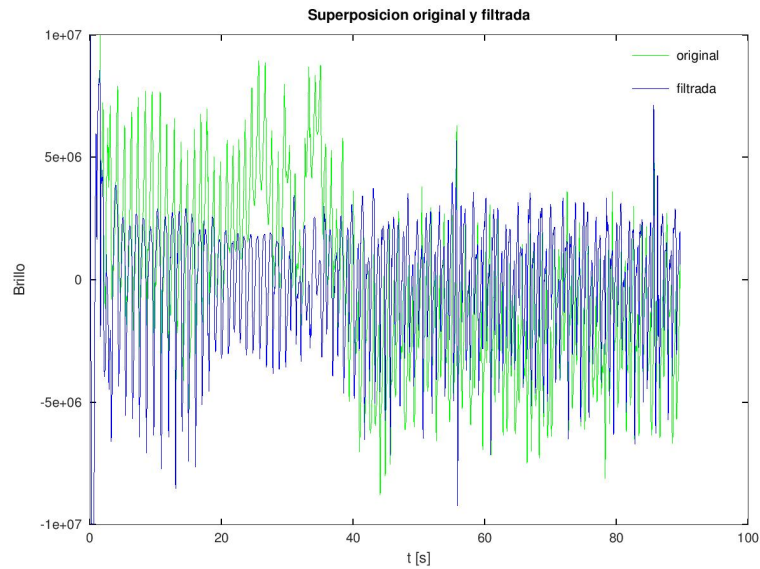


Figura 4.8 – Superposición de señales original y filtrada.

Como podemos notar se eliminó la componente de muy baja frecuencia sobre la cual estaba montada la señal original.

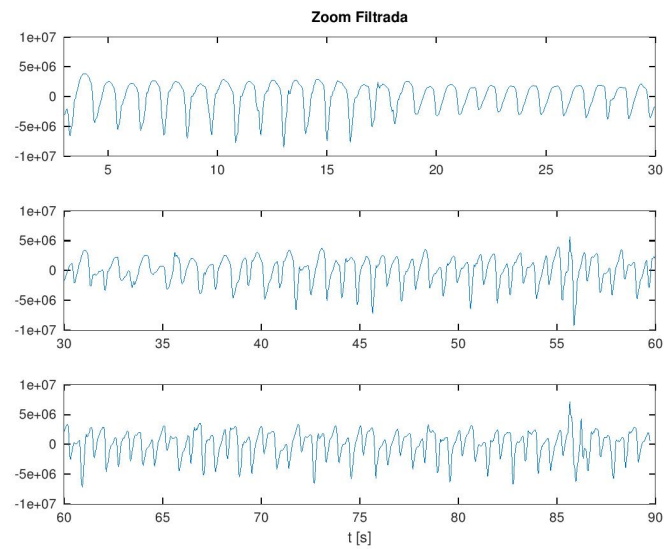


Figura 4.9 – Señal filtrada.

Sub items ejercicio 5

a) Teniendo una vista más detallada en la señal filtrada, se puede apreciar como hubo una remoción de derivas al no tener saltos sobre los valores promedios de la señal. Las frecuencias más bajas fueron removidas, la cual agregaba una componente senoidal muy lenta sobre la cual estaba montada la señal original.

b) Los cambios en la forma de la señal se deben a la deformación producida por la fase del filtro, debido a que no le afecta de igual forma a todas las frecuencias.

c) La señal filtrada tiene un retardo causado por la fase del filtro. Nos da una diferencia de 0.04 segundos.

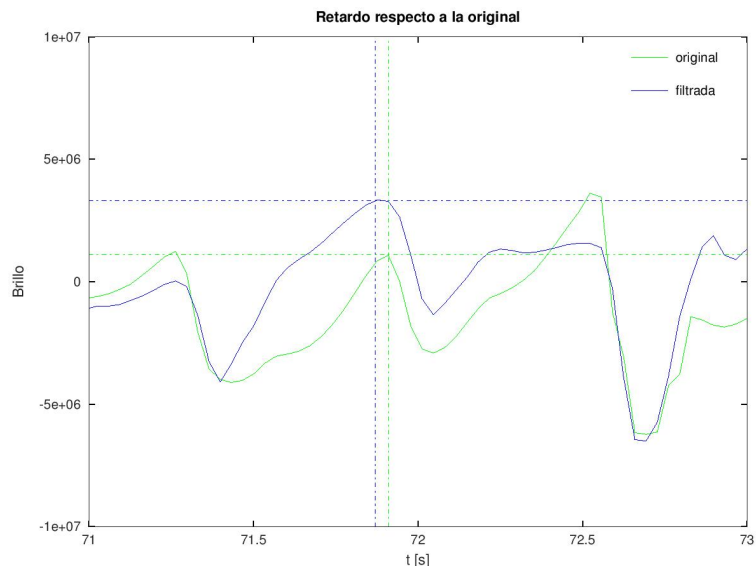


Figura 4.10 – Comparación del retardo de la señal original y filtrada.

4.6. Ejercicio 6.

Retardo de grupo.

$$Tg = \frac{-d\theta(\Omega)}{d\Omega} \quad (4.1)$$

Obtenemos una linealización del gráfico de la fase del filtro.

Finalmente concluimos que el retardo de fase es constante para todas las frecuencias de trabajo del filtro en banda plana.

Sin embargo no es posible decir lo mismo para las frecuencias que caen fuera de la banda plana, dando como resultado la deformación de la señal.

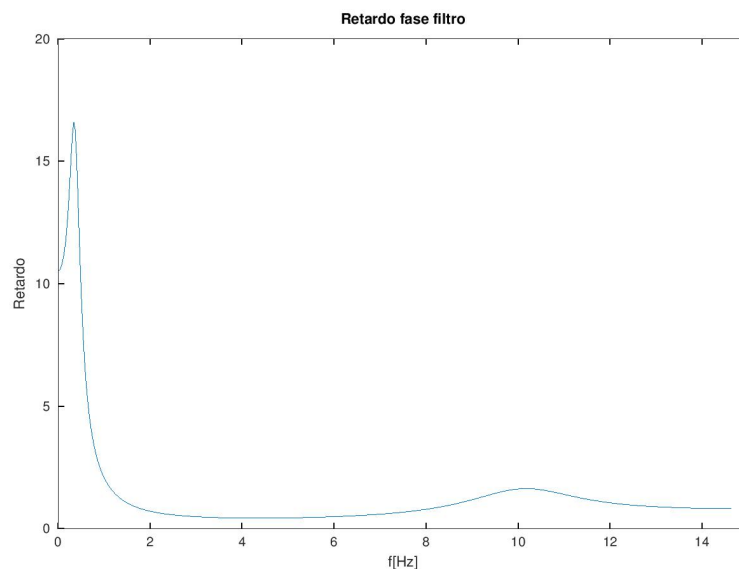


Figura 4.11 – Señal filtrada.

Como podemos ver para las frecuencias menores a la frecuencia de corte baja del filtro el retardo no es constante. Como tampoco para frecuencias mayores a la frecuencia de corte mayor del filtro.

Dentro del rango de frecuencias medias el retardo es constante.

$$tiempoSample = \frac{1}{frecSample} = \frac{1}{29,37Hz} = 0,03404s \quad (4.2)$$

$$retardoGrupo = 0,4 \quad (4.3)$$

$$0,4 * tiempoSample = 0,0136s \quad (4.4)$$

Comparamos con el resultado del punto 5, un retardo de fase de 0.04 segundos, por los resultados obtenidos del podemos ver que tiene un retardo de fase de 0.0136 segundos.

4.7. Ejercicio 7.

Se aplica la función **filtfilt** para el filtro diseñado anteriormente a la señal original. Dando como resultado el siguiente gráfico:

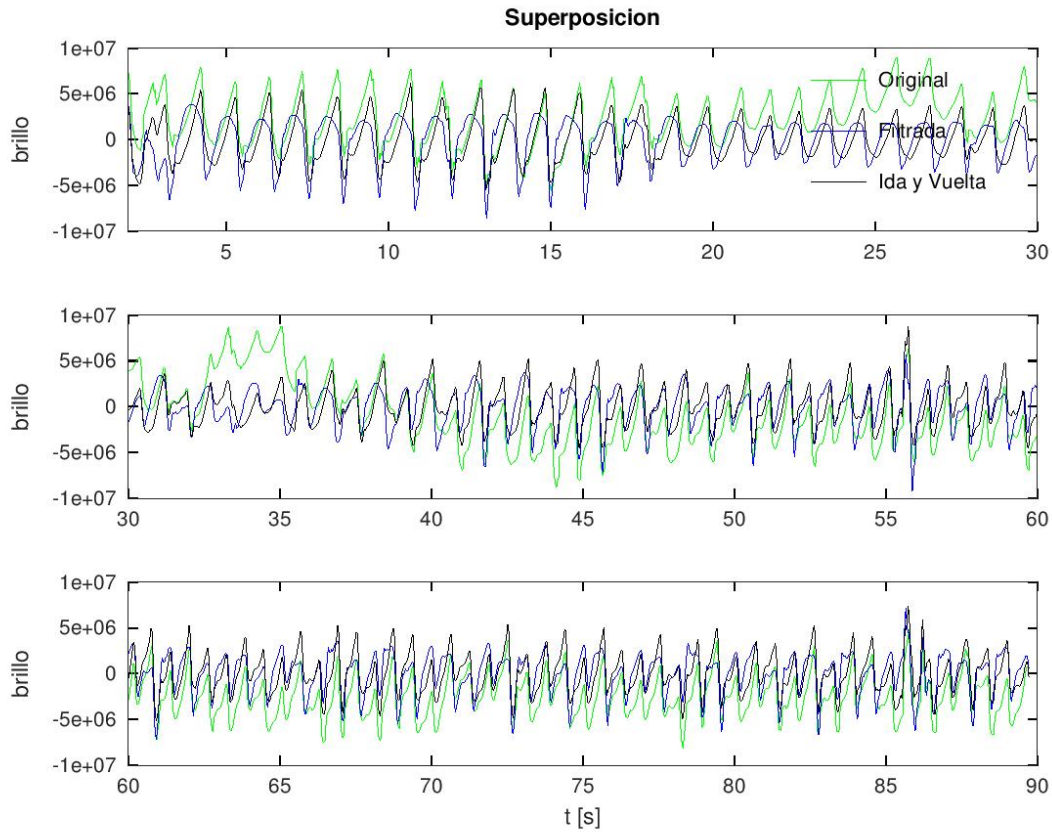


Figura 4.12 – Señales original, filtrada y filtrada con ida y vuelta.

Estamos viendo el resultado de hacer la operación de la señal pasando por el filtro y luego el filtro para transformado, dando como resultado una anulación del retardo de fase y el modulo al cuadrado.

De modo que la señal luego del filtro no sufre deformación debido a la fase, sin embargo afecta al cuadrado en modulo.

$$G(z) = H(z)H^*\left(\frac{1}{z^*}\right) \quad (4.5)$$

4.8. Ejercicio 8.

Se pidió hallar el espectrograma de la señal, yo propuse resolverlo con una ventana de Hamming de 128 puntos.

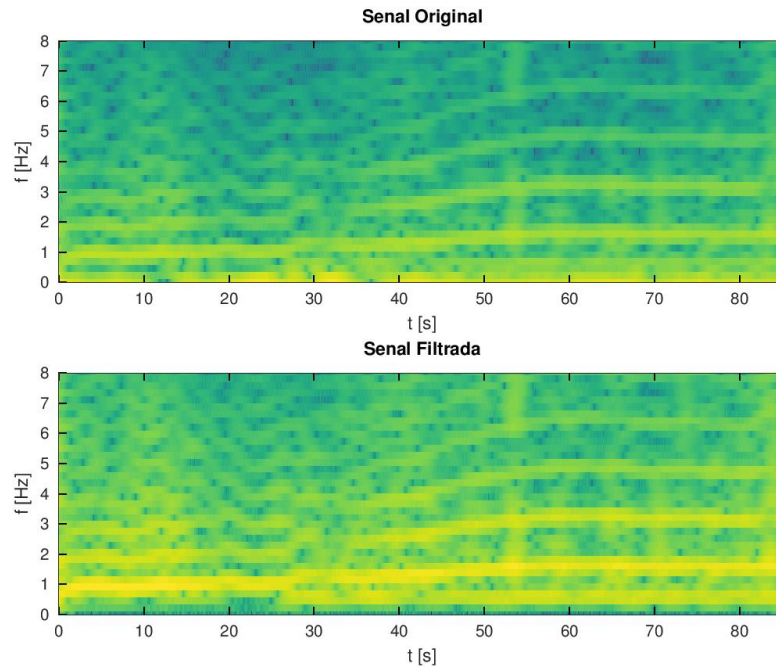


Figura 4.13 – Espectrograma de la señal original y la con ida y vuelta.

Se tuvo en cuenta la longitud de la ventana, ya que cuando más larga es, se pierde localización temporal pero gana localización en frecuencia.

Se utilizó una ventana de Hamming por tener el lóbulo principal ancho y el lóbulo secundario chico. De modo de tener una ventana suave.

A su vez esta ventana se desplazó en 1 punto obteniendo un solapamiento de 127 puntos, de manera de tener la máxima cantidad de muestreos en el espectrograma y poder ver en forma continua los cambios de frecuencia.

A diferencia del punto 2 y 3, ahora sí podemos notar la presencia de diferentes frecuencias para diferentes tiempos de la señal.

4.9. Ejercicio 9.

Podemos ver dos espectrogramas, el primero corresponde a la señal original, el segundo corresponde la señal filtrada.

La presencia energía de bajas frecuencias no nos permite ver claramente las frecuencias presentes en diferentes tiempos.

Finalmente en la segunda imagen, luego del paso de la señal a través del filtro pasabanda nos muestra como hay un primer cambio de ritmo en el segundo 20. Luego desde el segundo 40 al 60 se acelera el ritmo constantemente hasta alcanzar un ritmo cardíaco constante en el segundo 60 hasta el 90.

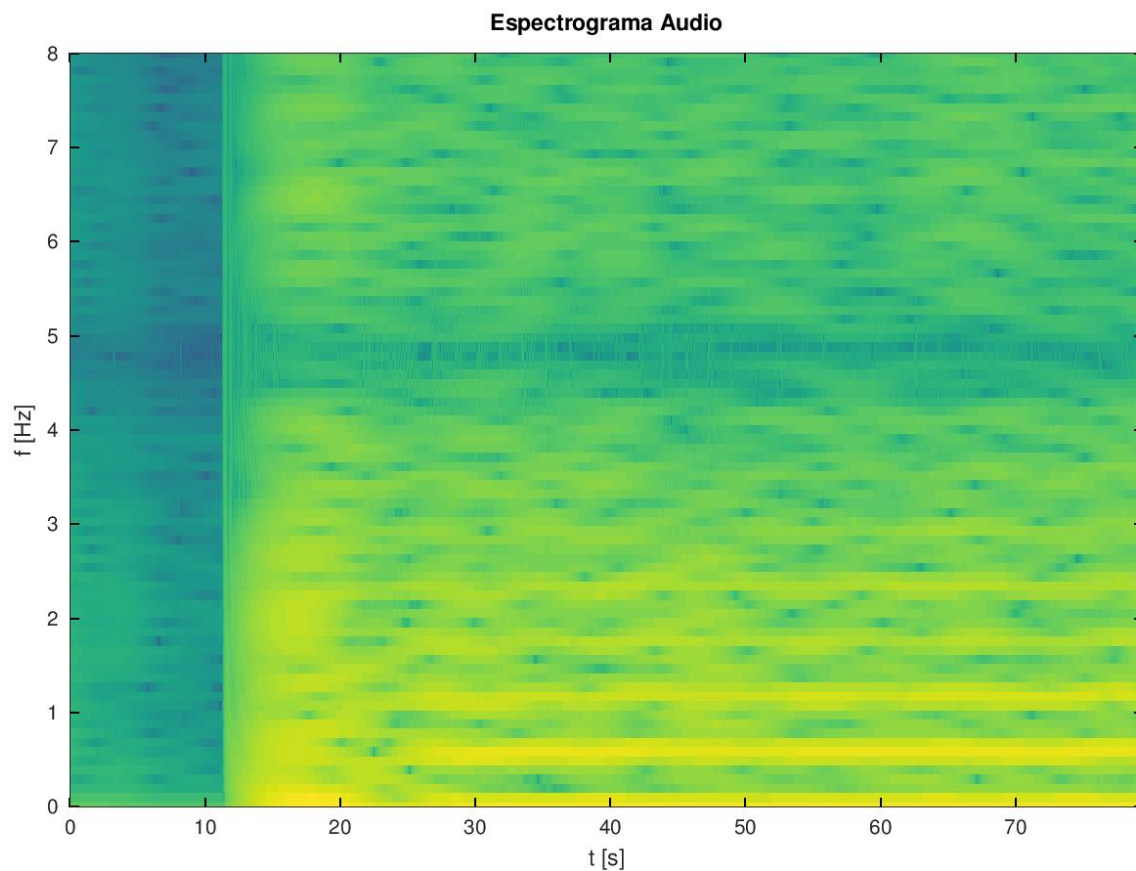


Figura 4.14 – Espectrograma de la señal de audio.

Viendo el resultado del espectrograma de audio podemos definir el origen de las frecuencias presentes en la muestra de video.

El audio influye en la muestra de video.

Debido a que inicia luego de 10 segundos y que tiene la misma componente y aparece superpuesta a la señal de audio podemos obtener dicha conclusión.

4.10. Ejercicio 10.

Se implementan las ultimas instancias para obtener los resultados buscados.

a) Se utiliza el filtro diseñado en el punto 7, para finalmente obtener el siguiente resultado.

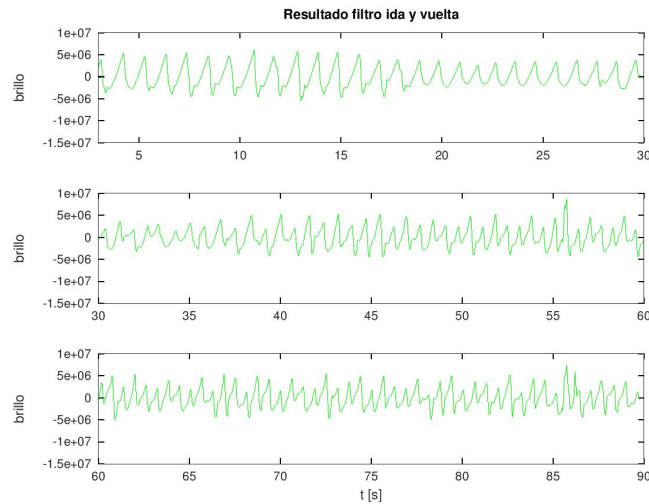


Figura 4.15 – Señal filtrada por el pasabanda.

b) Se implementa el filtro de derivada, dando como resultado lo siguiente.

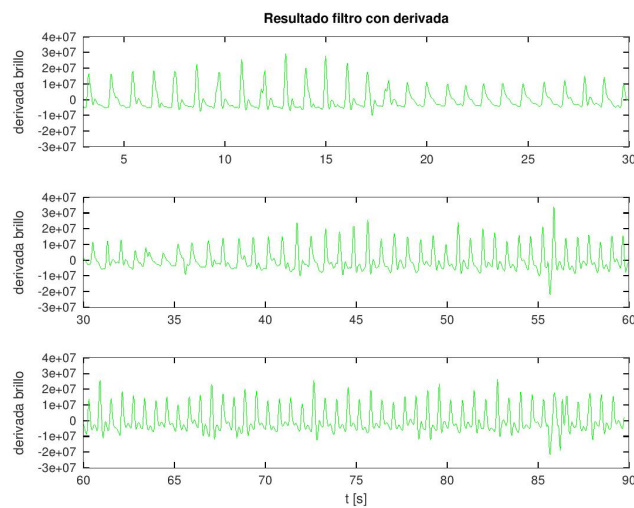


Figura 4.16 – Señal derivada por filtro.

La señal derivada se asemeja mucho más al pulso cardíaco.

c) Normalización con energía instantánea.

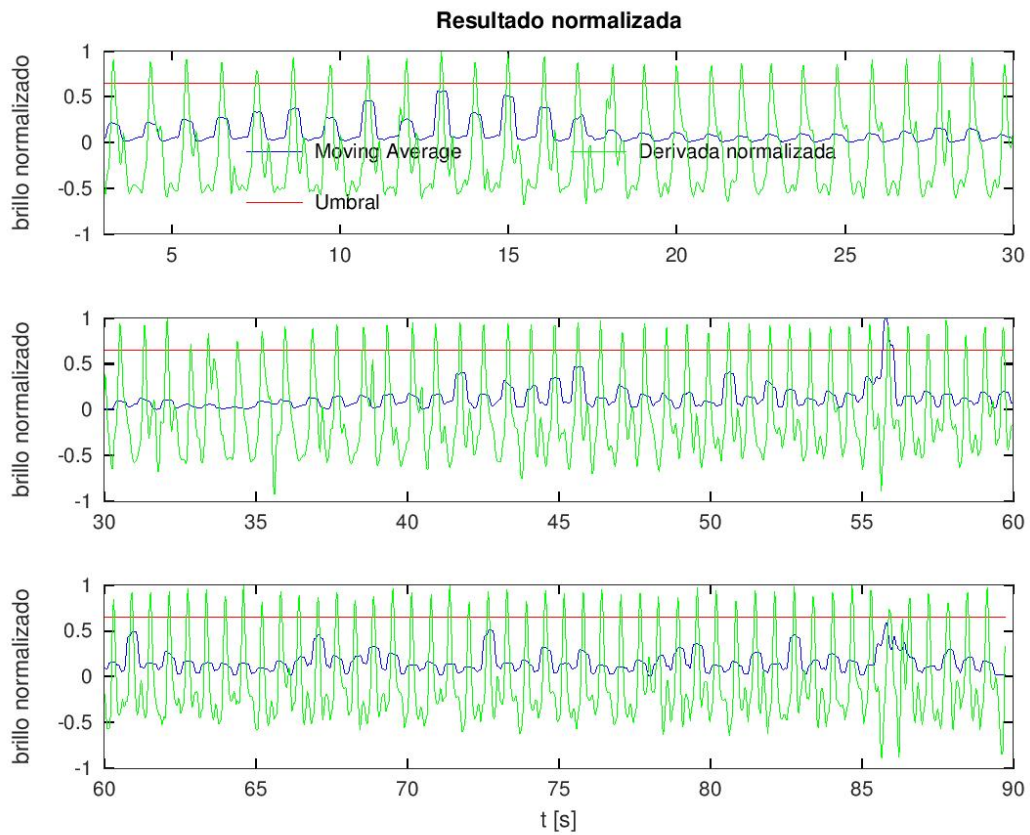


Figura 4.17 – Señal normalizada por la energía.

La utilidad de generar la normalización es para encerrar toda la señal sobre un nivel de alturas donde resulte más fácil realizar un algoritmo de búsqueda de picos.

d) Sobremuestreo con un factor de 4.

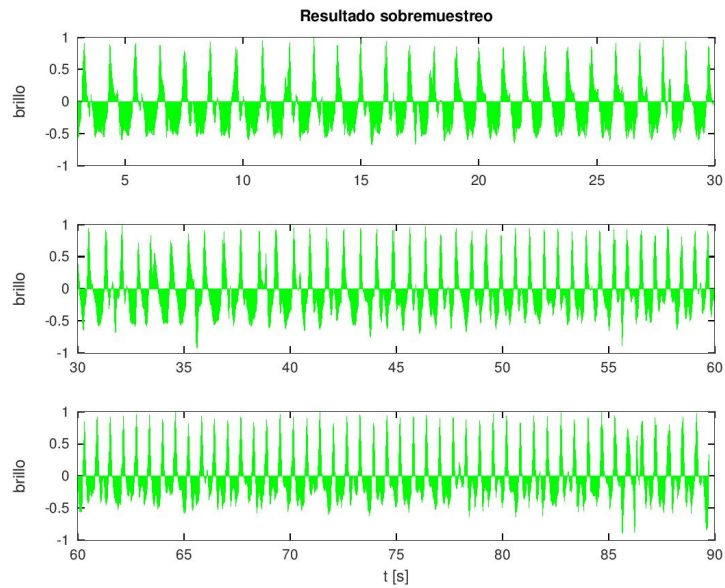


Figura 4.18 – Señal con ceros agregados.

Se realiza un sobremuestreo de factor 4 para poder obtener una mayor resolución temporal al momento de buscar por los picos de la señal utilizando un umbral de altura arbitraria.

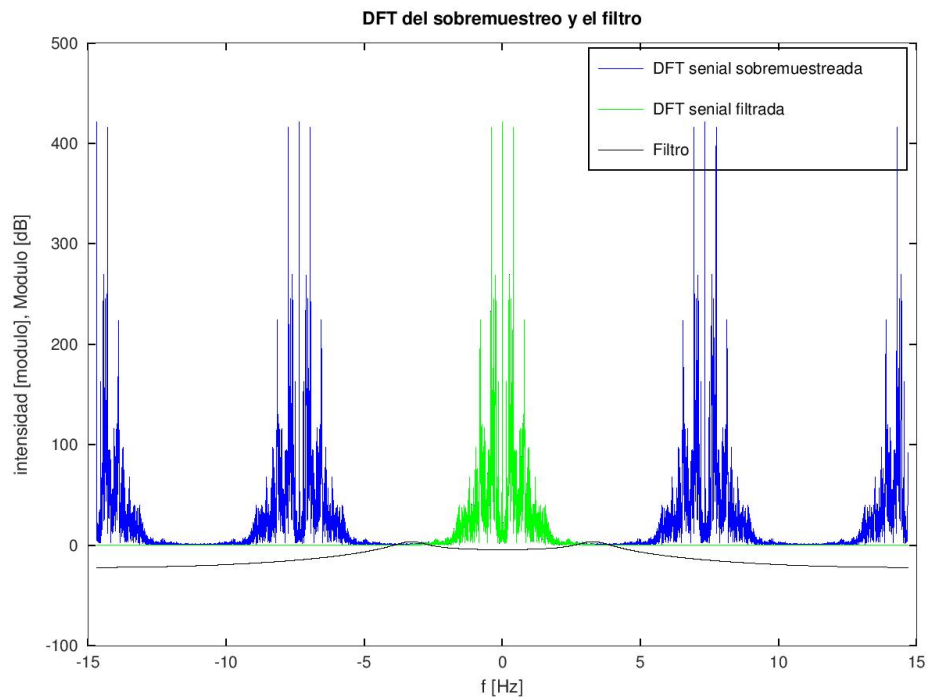


Figura 4.19 – DFT de la señal sobremuestreada, el filtro pasa bajo y la señal resultante.

Se representa la DFT de la señal sobremuestreada a la par de la respuesta en frecuencia del filtro para saber como seleccionar la frecuencia de corte baja, de manera de evitar aliasing y perder información, el orden del filtro pasa bajos es 7.

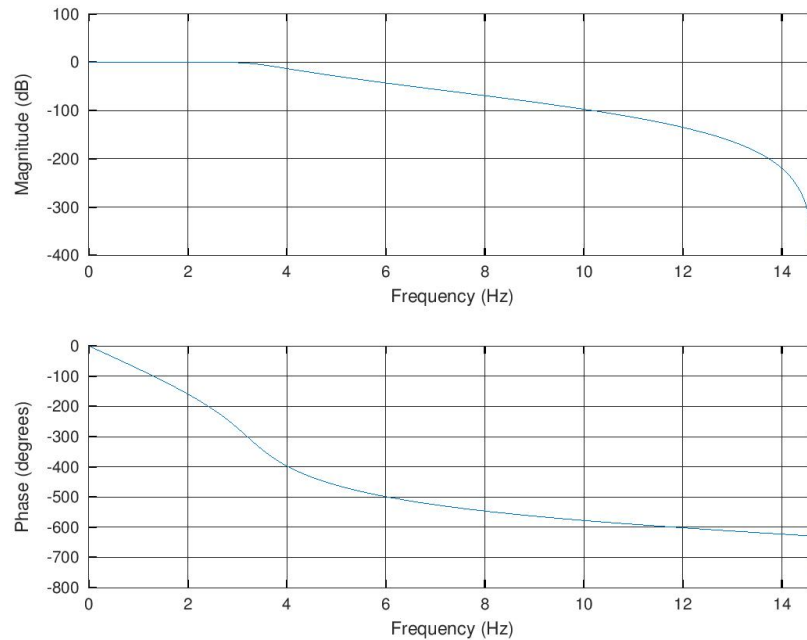


Figura 4.20 – Diagrama de Filtro modulo y fase.

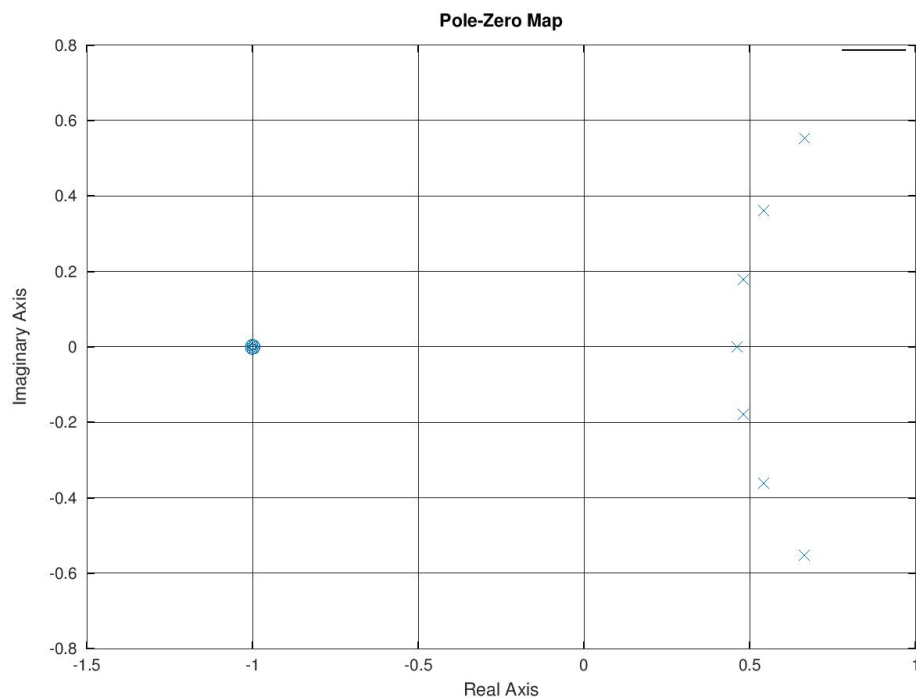


Figura 4.21 – Diagrama de polos y ceros.

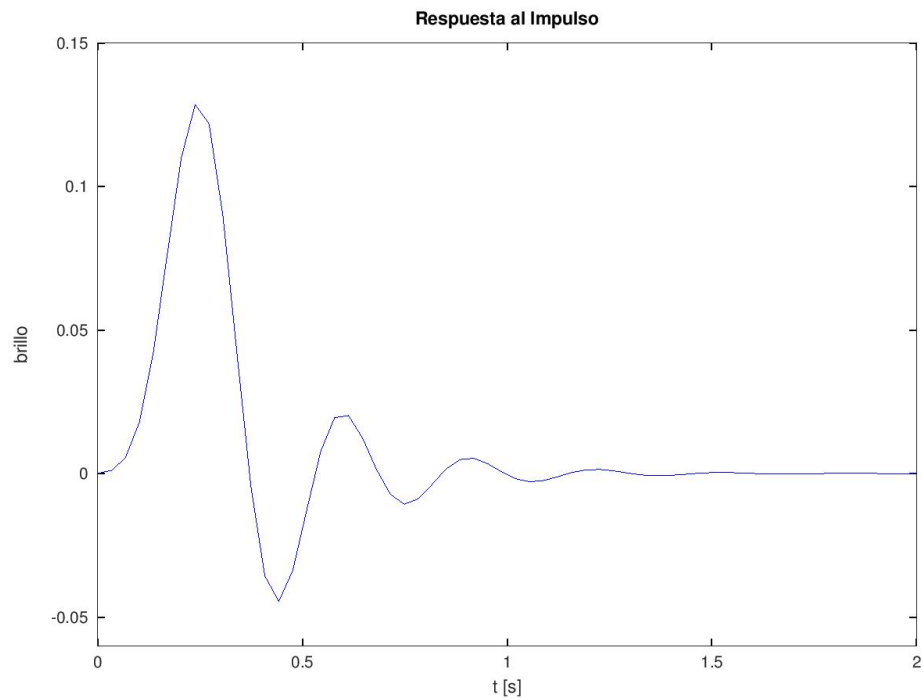


Figura 4.22 – Respuesta al impulso Filtro pasa bajo.

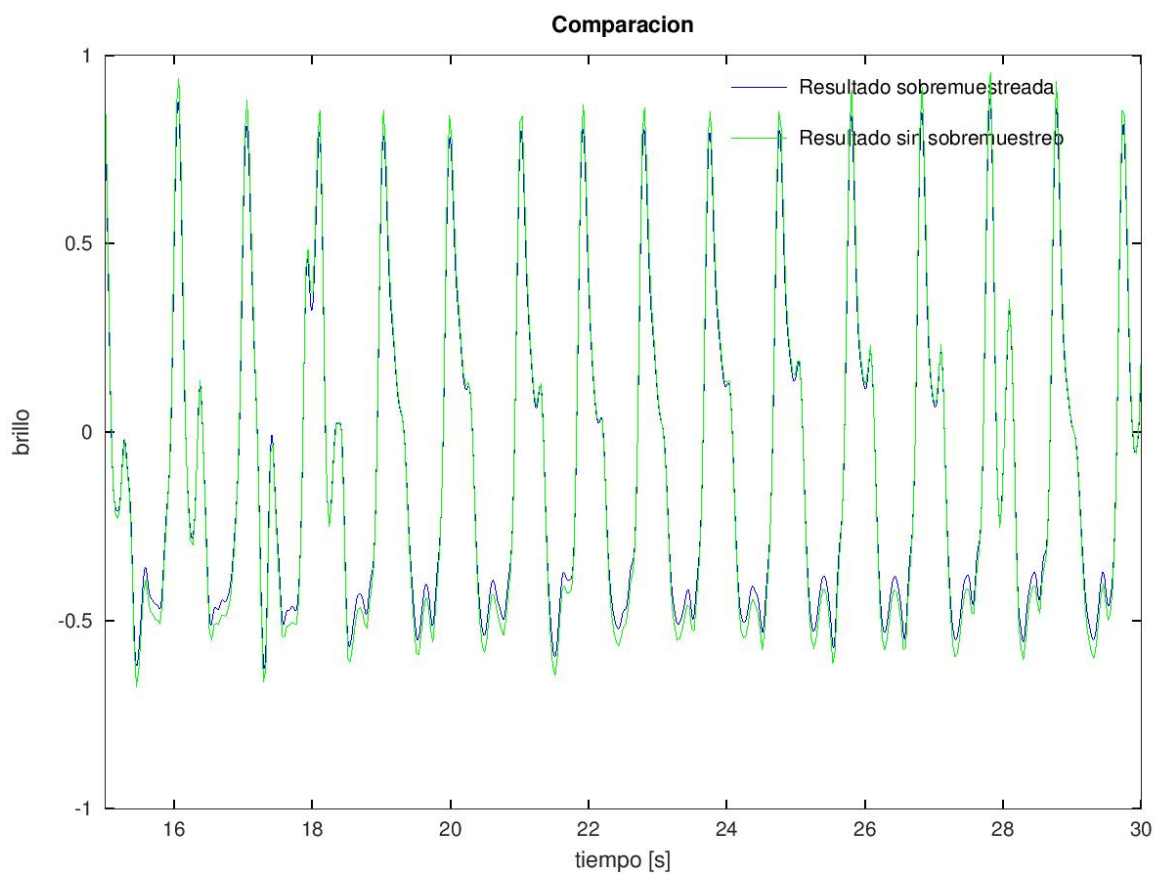


Figura 4.23 – Comparación de la señal y la señal sobremuestreada.

e) Se implementa un algoritmo de búsqueda de picos con un umbral establecido arbitrariamente en 65 % del máximo de la señal.

f)

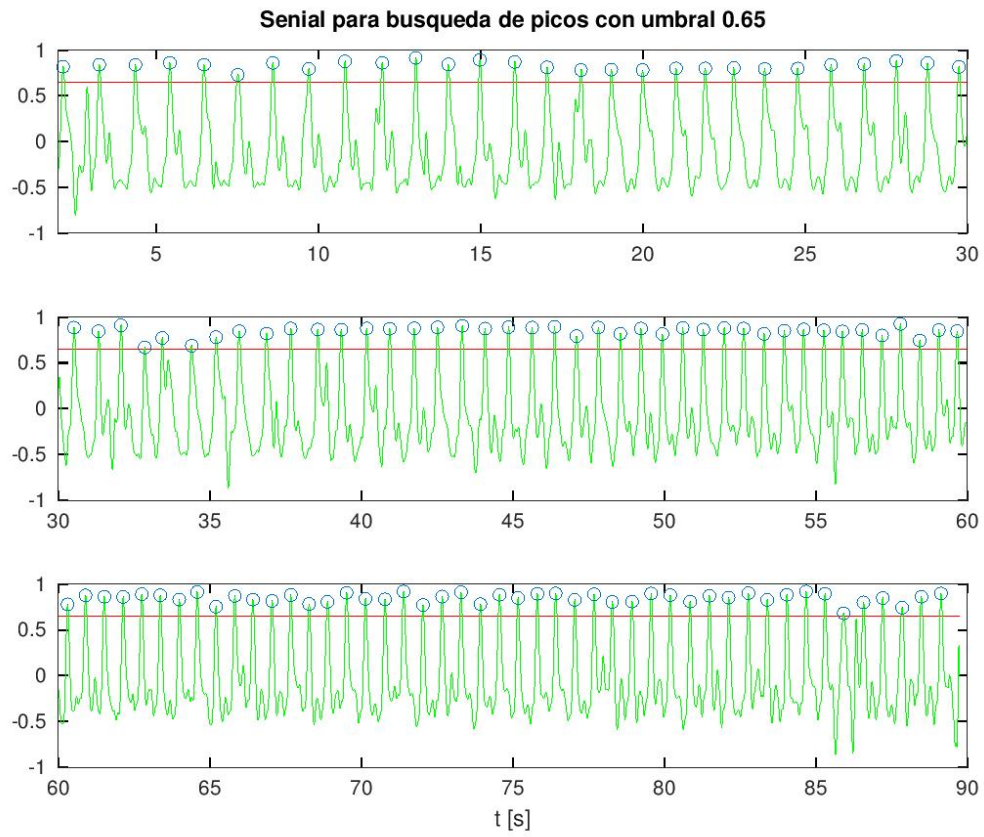


Figura 4.24 – Señal sobremuestreada, con el umbral y sus puntos de coincidencia.

4.11. Ejercicio 11.

Se obtiene los LPM de realizar la cuenta siguiente.

$$LPM = \frac{\text{latidos}}{\text{tiempo}} * 60 \quad (4.6)$$

Dando como resultado.

Intervalo [s]	LPM [latidos/min]
0-90	79
0-20	60
20-40	66
40-60	87
60-80	96
80-90	84

Como podemos ver el resultado general nos da un promedio, si tenemos en cuenta los diferentes intervalos podemos ver como varia las LPM.

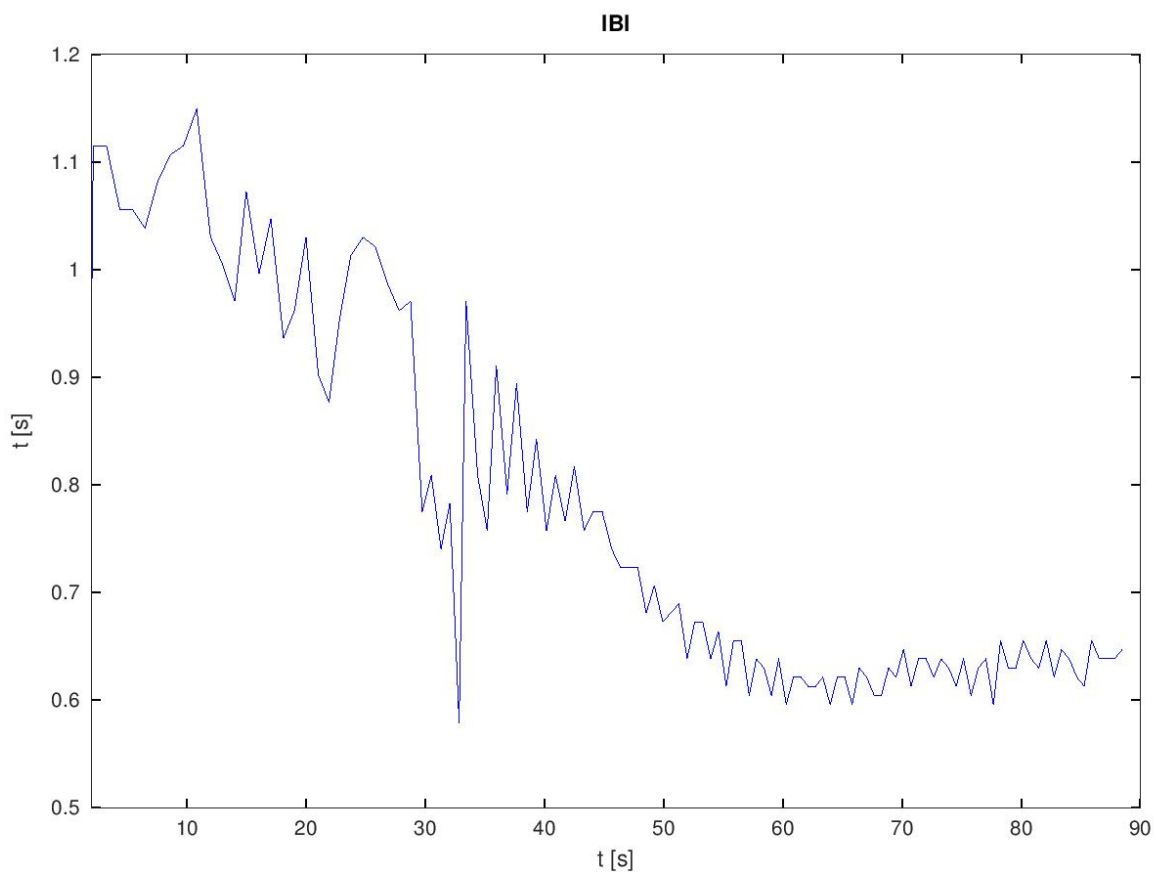


Figura 4.25 – Ibi.

El cuadro de IBI, muestra como la distancia entre los picos se va acortando cuando crece el tiempo.

4.12. Ejercicio 12.

- a) Se aplica el umbral a 200ms en la figura del IBI, no da ningún resultado ya que la menor diferencia en 580ms.
- b) Se implementa la mejora donde se haya dos picos con una relación de mas de 1.5 veces, se utiliza otro umbral. Finalmente hallamos nuevos picos para mejorar la lectura.

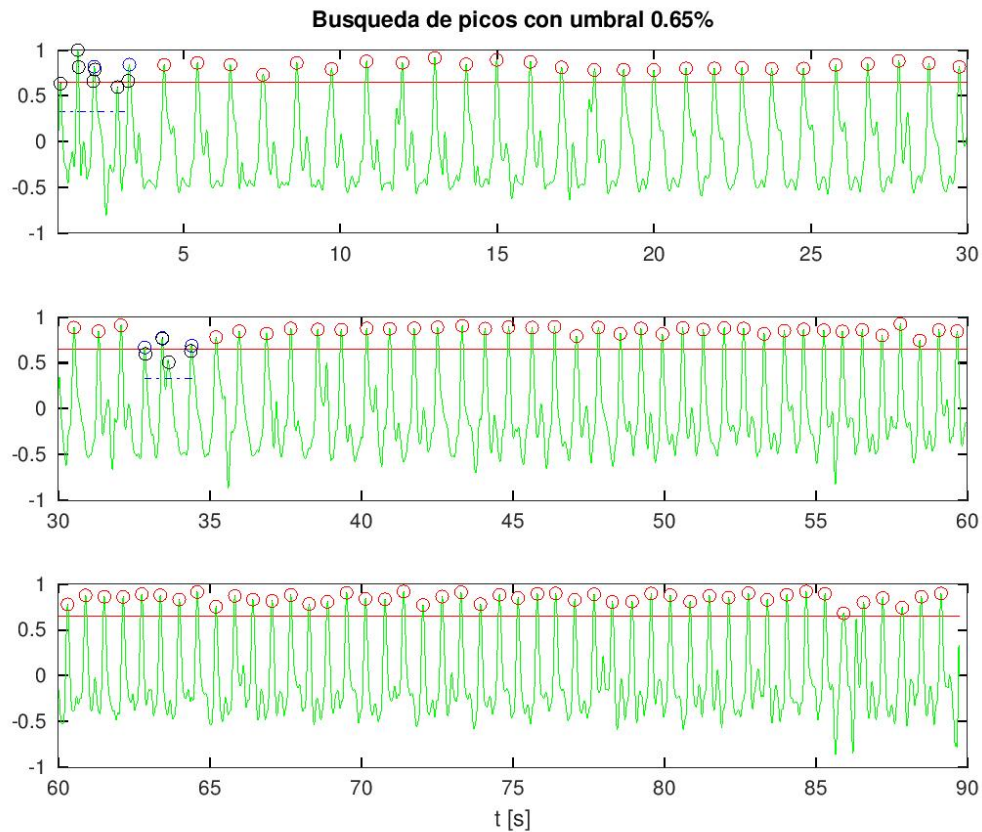


Figura 4.26 – Ibi.

4.13. Ejercicio 13.

Finalmente utilice la cámara de mi celular para obtener mi propia muestra.

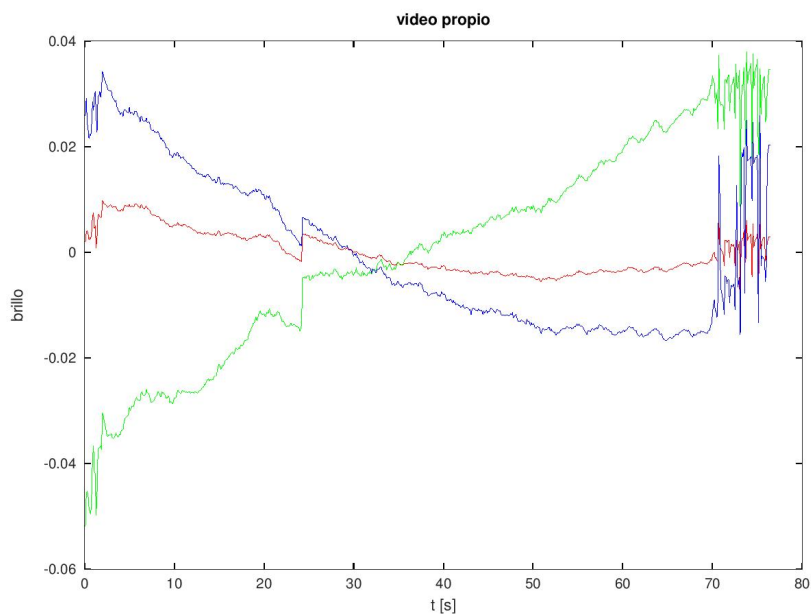


Figura 4.27 – Señales Rojas Verde y Azul en video propio.

Para comenzar no tengo seguridad que el video resulto ser una muestra indicada. Como podemos ver en el gráfico las señales no se asemejan en lo más mínimo a lo que teníamos originalmente.

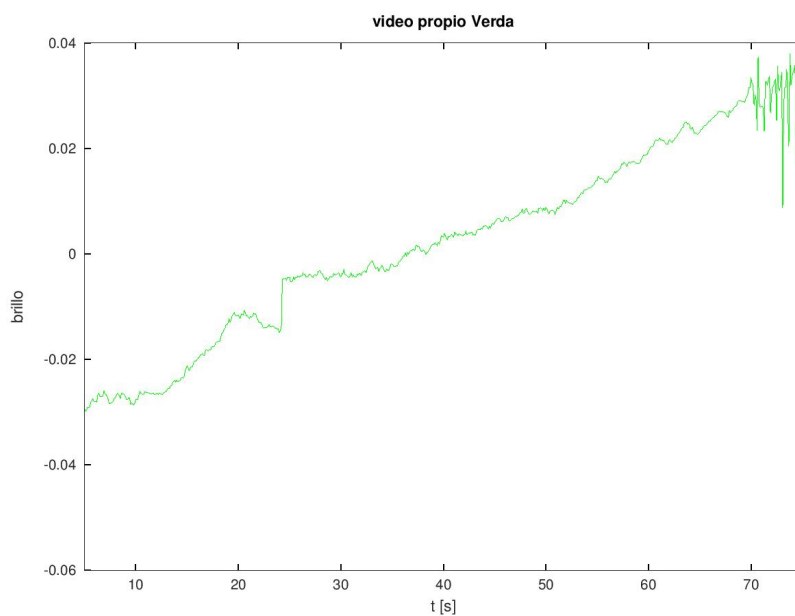


Figura 4.28 – Señal verde en video propio.

Seleccione la señal verde para continuar con la misma corriente de trabajo que utilizamos anteriormente.

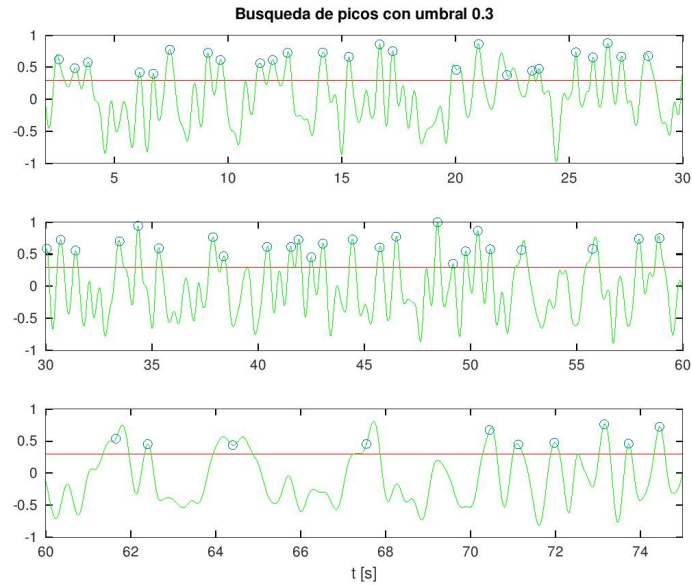


Figura 4.29 – Búsqueda de picos en video propio.

Como podemos notar la señal derivada y normalizada no posee, información clara, como lo hacían los datos originales, de modo que me vi forzado a disminuir el umbral de búsqueda de picos.

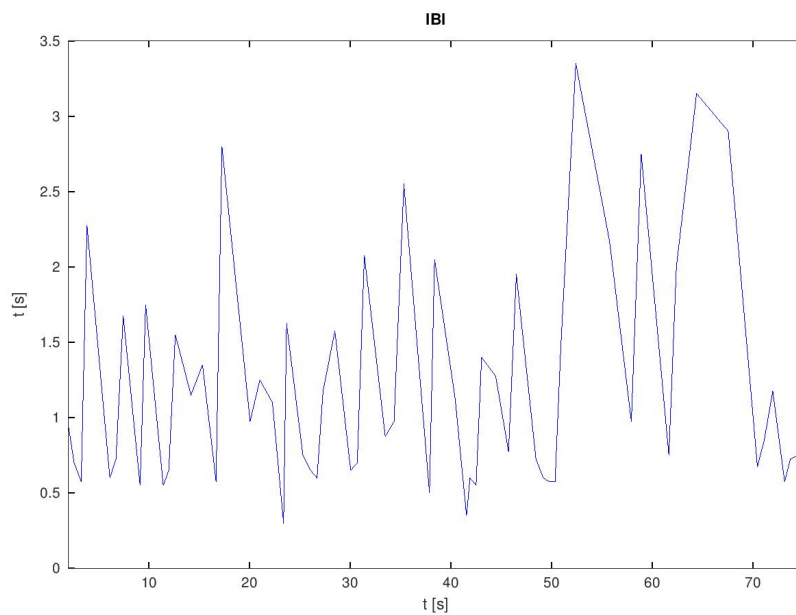


Figura 4.30 – Ibi video propio.

La figura del IBI nos informa que hay mucha diferencia entre dos picos consecutivos.

Para concluir obtuve los Latidos por Minuto.

Intervalo [s]	LPM [latidos/min]
0-75	50
0-20	51
20-40	54
40-60	51
60-75	40

Los resultados indican que mis latidos estan por debajo del promedio.

5. Conclusiones

Como conclusión podemos decir que es muy sencillo diseñar filtros digitales para procesar señales.

La correcta implementación de los filtros nos permiten, quitar frecuencias que no son de utilidad y ver con mayor exactitud un fenómeno físico que fue medido por medio de un método analógico.

El poder de procesamiento de una computadora combinado con el algoritmo correcto puede dar como resultado un programa eficiente para poder interpretar las señales que se encuentran presentes en la vida cotidiana.

De este modo, fue posible realizar una medición de los latidos por minuto de una persona sin necesidad de una herramienta de uso médico.