

Estudio de la Representación Módulo-Fase de la Transformada de Fourier en Tiempo Discreto en el Procesamiento Digital de Imágenes

Luciano Savoie¹, Ernesto Klimovsky¹, María Mercedes Gaitán¹

¹Grupo de Investigación en Enseñanza de la Matemática en Carreras de Ingeniería (GIEMCI)

Facultad Regional Paraná, Universidad Tecnológica Nacional

Almafuerte 1033, 3100, Paraná, Entre Ríos, Argentina

savoieluciano@gmail.com, erklimo@gmail.com, mgaitan@frp.utn.edu.ar

Resumen. Abordamos una propuesta de articulación de contenidos de tres asignaturas (Álgebra y Geometría Analítica, Análisis de Señales y Sistemas y Procesamiento Digital de Imágenes) de la carrera Ingeniería Electrónica de la Facultad Regional Paraná de la Universidad Tecnológica Nacional, en el marco del Proyecto de investigación “Aportes de Matemática aplicada en Ingeniería sustentados en elementos de Álgebra Lineal”, que intenta mostrar a los estudiantes la importancia de los conceptos matemáticos adquiridos en el Ciclo Básico para resolver situaciones concretas de Ingeniería.

En este trabajo vinculamos la modelización matemática de las imágenes digitales como matrices, la representación Módulo-Fase de la transformada discreta bidimensional de Fourier y exponemos su aplicación concreta en el Procesamiento Digital de Imágenes.

Comenzamos con los conceptos matemáticos necesarios, desarrollados en las asignaturas antes mencionadas, exponemos aquellos relacionados al tratamiento de imágenes digitales, realizamos simulaciones mediante software, mostramos ejemplos prácticos y mencionamos aplicaciones.

Palabras Clave: Modelización matemática, Transformada de Fourier, Procesamiento de imágenes.

1 Introducción

Un desafío que enfrentamos los docentes universitarios del área Matemática es qué estrategia usar para enseñar determinado tema, logrando motivar al alumno y que dicho aprendizaje desarrolle las competencias profesionales que perduren en el tiempo. El uso de una estrategia metodológica referida a resolución de problemas que integre conceptos de distintas asignaturas permite contextualizar los contenidos, facilitando su aprendizaje. En este caso, trabajamos en la articulación de tres asignaturas de la carrera de la siguiente manera:

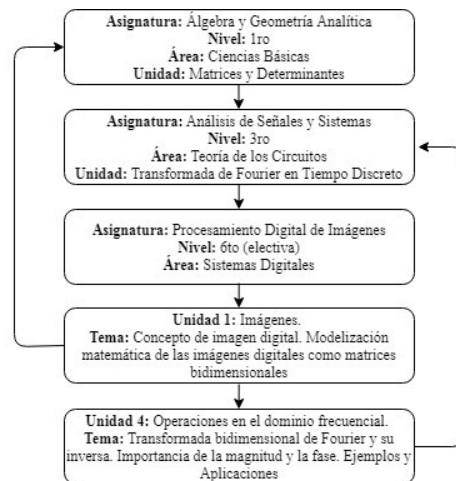


Fig. 1. Esquema de articulación de asignaturas planteado, detallando las unidades relacionadas y la vinculación entre ellas.

Tenemos presentes los documentos elaborados por el Consejo Federal de Decanos de Facultades de Ingeniería (CONFEDI) de la República Argentina, en particular el correspondiente al año 2014 denominado *Competencias en Ingeniería*, en el cual se expresa *Competencia es la capacidad de articular eficazmente un conjunto de esquemas (estructuras mentales) y valores, permitiendo movilizar (poner a disposición) distintos saberes, en un determinado contexto con el fin de resolver situaciones profesionales.* [1]

Consideramos que resolver problemas relativos a la especialidad elegida, abordando y vinculando los contenidos de las diversas asignaturas e incorporando tecnología al aula, posibilita de alguna manera superar el desafío planteado inicialmente con el fin de lograr las competencias requeridas a un ingeniero, que incluyen entre otras: la identificación, la formulación y el análisis del problema, la búsqueda de soluciones y la selección criteriosa de la alternativa más adecuada.

2 Marco teórico

2.1 Imágenes digitales

Una imagen puede ser definida como una función bidimensional $f(x,y)$, donde x e y son las coordenadas en el plano y f es la amplitud de cualquier par de coordenadas (x,y) , llamada intensidad de la imagen en el punto.

El término nivel de gris se usa para referirse a la intensidad monocromática de las imágenes. Por su parte las imágenes a color están formadas por una combinación individual de imágenes 2-D. En el sistema de color RGB, un color consiste en tres componentes individuales de una imagen que son Red, Green y Blue (rojo, verde y azul respectivamente). Por esta razón, muchas técnicas de desarrollo para imágenes monocromáticas pueden ser extendidas para el procesamiento de imágenes a color.

Cuando (x, y) y los valores de la amplitud de la función f son cantidades discretas finitas, a dicha imagen se le llama imagen digital. Una imagen digital está compuesta de un número finito de elementos llamados píxeles, donde cada uno tiene una localidad y un valor establecido. [2]

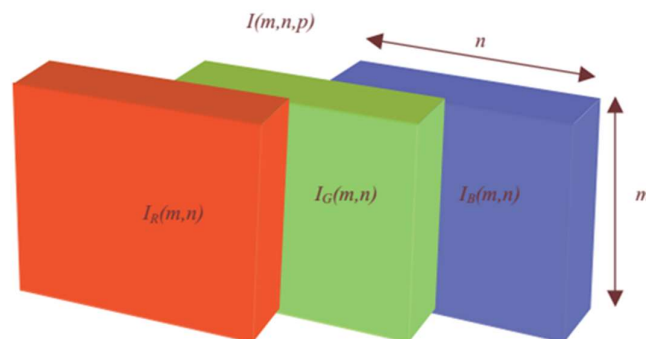


Fig. 2. Esquema de una imagen RGB $I(m,n,p)$ con sus tres componentes individuales. Aquí m y n tienen el mismo significado que para el caso de las imágenes monocromáticas, mientras que p representa el plano, que para RGB es 1 para el rojo, 2 para el verde y 3 para el azul. [3]

2.2 Manejo de imágenes mediante software

Ya sea que utilicemos MatLab o una alternativa en software libre como SciLab, la estructura básica de datos es el arreglo, el cual se puede definir como un conjunto ordenado de números reales o complejos. En el caso de las imágenes, éstas pueden ser representadas por matrices formadas por conjuntos ordenados de valores reales que, a su vez, representan la intensidad de color o de niveles de grises.

Estos programas almacenan las imágenes como arreglos bidimensionales (matrices) en los cuales cada elemento de la matriz corresponde a la intensidad de un píxel de la imagen. Por ejemplo, una imagen de 200 filas por 300 columnas se almacena como una matriz de 200×300 . Algunas imágenes, como las RGB nombradas anteriormente, requieren de un arreglo tridimensional, donde el primer plano representa la intensidad de rojo en los píxeles, el segundo plano, la intensidad de verde de los píxeles, y el tercer plano, la intensidad de azul de ellos.

Esta convención sustituye el trabajo con imágenes por una representación matricial. Por ejemplo, se puede seleccionar un solo píxel de una imagen-matriz (estos términos serán análogos a partir de ahora para las definiciones que siguen) de la forma $I(2,15)$, con lo cual el software regresará el valor del píxel localizado en el renglón 2, columna 15 de I . [2]

2.3 Transformada de Fourier en tiempo continuo y su inversa

La *Transformada de Fourier* de una función continua unidimensional $f(x)$ viene dada por la expresión:

$$F(u) = \int_{-\infty}^{\infty} f(x) \cdot e^{-j2\pi u x} dx \quad (1)$$

donde u es la variable en el espacio de la frecuencia, y $j = \sqrt{-1}$.

Análogamente, dada $F(u)$, puede obtenerse la función $f(x)$ calculando la *Transformada Inversa de Fourier*:

$$f(x) = \int_{-\infty}^{\infty} F(u) \cdot e^{j2\pi u x} du \quad (2)$$

Estas dos ecuaciones componen lo que denominamos *Par de Transformadas Fourier*.

La *Transformada de Fourier* de una función consta, en general, de una parte real $R(u)$ y otra imaginaria $I(u)$, de forma que podemos expresar $F(u)$ en coordenadas rectangulares:

$$F(u) = R(u) + jI(u) \quad (3)$$

o también puede expresarse en términos de amplitud $|F(u)|$ y fase $\phi(u)$ (coordenadas polares):

$$F(u) = |F(u)| \cdot e^{-j\phi(u)} \quad (4)$$

donde

$$|F(u)| = \sqrt{R^2(u) + I^2(u)} \quad (5)$$

$$\phi(u) = \tan^{-1} \left[\frac{I(u)}{R(u)} \right] \quad (6)$$

A $|F(u)|$ se le conoce normalmente como el *Espectro de Fourier* o *Módulo de la Transformada de Fourier*, y a $\phi(u)$ como su *Ángulo de Fase* o simplemente, *Fase de la Transformada de Fourier*.

El término exponencial compleja de la *Transformada Inversa de Fourier* puede escribirse, siguiendo el Teorema de Euler, de la siguiente forma:

$$e^{j2\pi u x} = \cos 2\pi u x + j \cdot \sin 2\pi u x \quad (7)$$

lo que nos permite ver que la función original $f(x)$ está compuesta por un número infinito de funciones seno y coseno, donde el valor de u representa la frecuencia de los respectivos pares seno y coseno.

La *Transformada de Fourier* puede extenderse a una función bidimensional $f(x, y)$:

$$F(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \cdot e^{-j2\pi(ux+vy)} dx dy \quad (8)$$

y su transformada inversa es:

$$f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u, v) \cdot e^{j2\pi(ux+vy)} dudv \quad (9)$$

donde u y v son las dos variables de frecuencias. El módulo y la fase pueden calcularse de forma análoga al caso unidimensional. [4]

2.4 Transformada de Fourier en tiempo discreto y su inversa

Suponiendo funciones bidimensionales de variables discretas, como es el caso de las imágenes digitales, la *Transformada Discreta de Fourier* o *TDF* se obtiene empleando sumatorias en lugar de integrales:

$$F(u, v) = \frac{1}{M \cdot N} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \cdot e^{-j2\pi(\frac{ux}{M} + \frac{vy}{N})} \quad (10)$$

donde M representa el número de columnas (píxeles en la dirección X) y N el número de filas (píxeles en la dirección Y). La *Transformada Inversa* viene dada por la expresión:

$$f(x, y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) \cdot e^{j2\pi(\frac{ux}{M} + \frac{vy}{N})} \quad (11)$$

El módulo y la fase se calculan de la misma forma que en el caso de una función continua, con la única diferencia de que las variables independientes son ahora discretas:

$$|F(u, v)| = \sqrt{R^2(u, v) + I^2(u, v)} \quad (12)$$

$$\phi(u, v) = \tan^{-1} \left[\frac{I(u, v)}{R(u, v)} \right] \quad (13)$$

donde $R(u, v)$ y $I(u, v)$ son las partes real e imaginaria de $F(u, v)$ respectivamente. [4]

3 Implementación en software

3.1 Módulo y Fase de la Transformada Discreta de Fourier (TDF) de una imagen digital

Una vez expuestos los conceptos matemáticos necesarios para comprender la TDF en tiempo discreto y su vinculación con las imágenes digitales, continuamos con la implementación computacional mediante el paquete MatLab. Este nos permite realizar la TDF, expresarla en términos de magnitud y fase, además, poder representar gráficamente a ambas.

Por una cuestión práctica y para facilitar la comprensión de los temas, a continuación, mostramos ejemplos desarrollados sobre imágenes monocromáticas de 8 bits. En ellas, el valor de cada píxel se corresponde a un determinado valor en una escala de grises de 2^8 , es decir, 256 tonalidades distintas. La misma está comprendida en un rango de números enteros de 0 a 255 en el cual cada valor está asociado a distintas graduaciones de gris, desde el negro cuyo valor es el 0, hasta el blanco cuyo valor es 255. Todos los conceptos desarrollados pueden hacerse extensivos a imágenes color, recordando que éstas son tratadas como arreglos tridimensionales, como se mencionó anteriormente. [2]

Un ejemplo de imagen monocromática y su correspondiente representación matricial se presenta en la **Fig. 3**

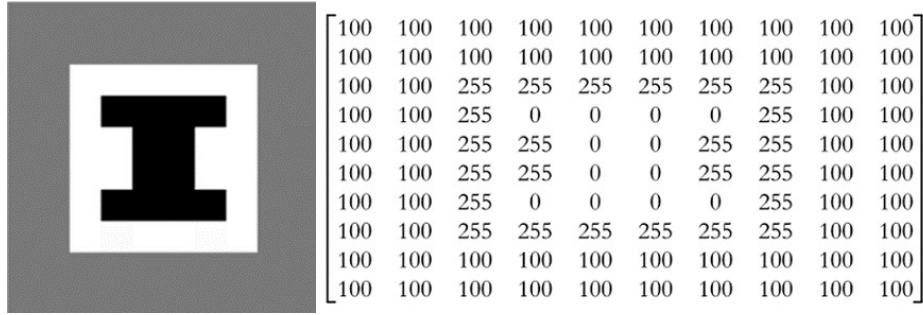


Fig. 3. Representación de una imagen en escala de grises mediante una matriz. [5]

Un número elevado de niveles de gris se corresponderá con una imagen que se representará digitalmente con un mayor número de bits y de la misma forma tendrá la capacidad de representar imágenes con mayor *rango dinámico*. El rango dinámico, expresado en Decibeles (dB), se define matemáticamente como:

$$DB = 10 \log \frac{(Amplitud\ Máxima)^2}{(Amplitud\ Mínima)^2} \quad (14)$$

En un sistema digital se considera la amplitud máxima como 2^n , siendo n el número de bits, y como amplitud mínima 2^0 (es decir, 1), correspondiente al bit 0. [6]. Por ejemplo, un digitalizador de 8 bits puede desplegar 256 niveles de gris y corresponderá a un sensor con un rango dinámico de 48 dB.

Tomando como ejemplo la imagen “cameraman.tif” (una de las imágenes de prueba en escala de grises más populares, de tamaño 256×256) realizamos una serie de instrucciones resumidas en la Fig.4, obteniendo los resultados expuestos en la Fig. 5.

Para el cálculo práctico de la TDF (y su inversa) de una imagen, Matlab emplea un algoritmo basado en la Transformada Rápida de Fourier o FFT (Fast Fourier Transform) [7]. Este método supone un gran ahorro de cantidad de cálculos realizados y tiempo de procesado.

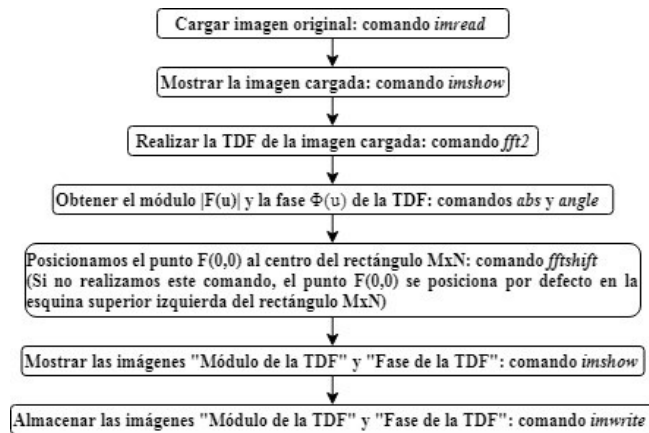


Fig. 4. Etapas del procesamiento realizado en MatLab destacando los comandos utilizados para la obtención de la TDF, expresarla en términos de módulo y fase, y poder representar gráficamente a ambas. La visualización de la imagen “Módulo de la TDF” se realiza aplicando la función $\log(1 + |F(u,v)|)$ en lugar de $|F(u,v)|$.

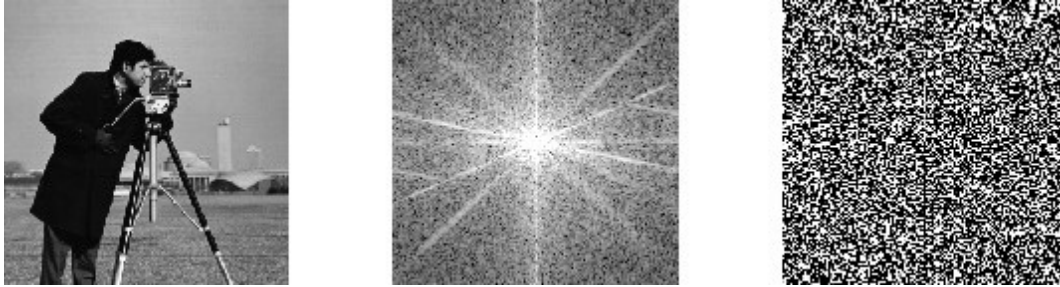


Fig. 5. Izquierda: Imagen utilizada como ejemplo “cameraman.tif”. Centro: Imagen “Modulo de la TDF”. Derecha: Imagen “Fase de la TDF”. Todas ellas obtenidas a partir de la implementación del procesamiento detallado en la **Fig.4**

Hay que superar algunas dificultades al mostrar el espectro de frecuencia de una imagen. La primera surge debido al amplio rango dinámico de los datos resultantes de la TDF. En la imagen original, el valor de un píxel será un número entero entre $[0,255]$, representando el grado de intensidad, pero en la imagen que representa el módulo de Fourier los valores de los píxeles son números en punto flotante y no están limitados a los valores de $[0,255]$. Estos datos deben ser escalados de nuevo para transformarlos en un formato visible, de forma que no exceda la capacidad del dispositivo de visualización. Una cuantización lineal simple no proporciona siempre los mejores resultados, pues muchas veces se pierden los puntos de baja amplitud. Una solución común a este problema es representar el logaritmo del espectro, mejor que el espectro por sí mismo. De esta forma, la función logarítmica realiza una compresión del rango dinámico, facilitando considerablemente su visualización y su interpretación. [6]

A primera vista, las imágenes “Modulo de la TDF” y “Fase de la TDF” no parecen proporcionarnos demasiada información respecto a la imagen original y pueden parecernos conceptos abstractos. A partir de esto, planteamos las siguientes preguntas: ¿Es más importante el módulo o la fase de una imagen? ¿cuál de las dos contiene más información de la imagen? Al momento de restaurar una imagen ¿cuál componente (módulo o fase) es más importante o con cuál de las dos es más importante contar? El análisis realizado a continuación nos permitirá resolver estas cuestiones y ver la importancia que tiene cada componente.

3.2 Imagen de “Solo Módulo” e Imagen de “Solo Fase”

Como vimos, cualquier señal unidimensional o bidimensional expresada por medio de su *Transformada de Fourier*, puede ser recuperada completamente por el proceso inverso sin pérdida de información. Es decir, podemos ir y volver del espacio de frecuencias al (x,y) sin perder información. A través del comando *iffi2* podemos obtener la inversa de la TDF [7], este nos permite realizar un proceso de reconstrucción de una imagen a partir de sus componentes de módulo y fase.

Un método utilizado para exponer la importancia de las componentes que conforman la TDF, consiste en realizar un proceso de reconstrucción de una imagen a través de la inversa de la TDF, pero conservando una componente original de la misma (módulo o fase) y modificando la restante a un valor fijo determinado, para así poder observar cuál de las dos tiene un papel preponderante, dando origen a los siguientes tipos de imágenes:

- *Imagen de solo módulo*: La TDF tiene como módulo el de la imagen original, y fase nula.
- *Imagen de solo fase*: La TDF tiene módulo unitario y la fase igual al de la imagen original.

Tomando nuevamente como ejemplo la imagen “cameraman.tif” realizamos una serie de instrucciones resumidas en el diagrama de la **Fig.6**, para obtener las imágenes de “Solo Módulo” y “Solo Fase”, presentadas en la **Fig. 7**.

En teoría, si la imagen original es real, entonces el resultado del comando *iffi2* debería ser real. En la práctica, sin embargo, el resultado de *iffi2* a menudo tiene componentes imaginarios muy pequeñas como resultado de errores de redondeo que son característicos de los cálculos en coma flotante. Por lo tanto, una buena práctica es extraer la parte real del resultado (comando *real*) después de calcular la inversa de la TDF para obtener una imagen que conste solo de valores reales. [7]

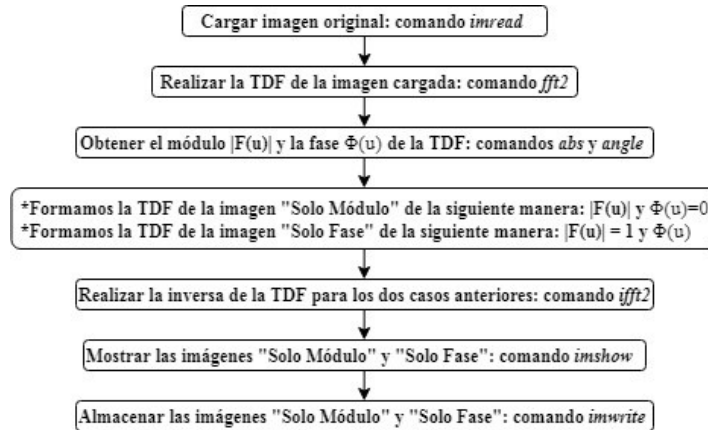


Fig. 6. Etapas del procesamiento realizado en MatLab destacando los comandos utilizados para la obtención de las imágenes “Solo Módulo” y “Solo Fase”.

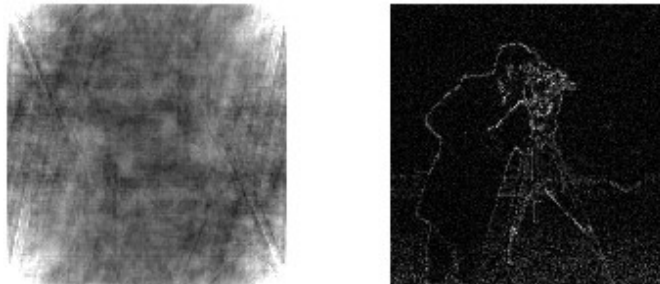


Fig. 7. Izquierda: Imagen “Solo Módulo”. Derecha: Imagen “Solo fase”, obtenidas a partir de la implementación del procesamiento detallado en la Fig.6.

Observamos que muchas de las características de la imagen original son identificables claramente en la imagen de “Sólo fase”, pero no ocurre lo mismo en la imagen de “Sólo módulo”, lo que nos permite deducir que gran parte de la información se encuentra codificada en la fase. Las características principales de la imagen original son identificables en una imagen de “Sólo fase”, ya que la inteligibilidad está asociada a los detalles (puntos, bordes, etc.), como dicha imagen ilustra en la Fig. 7. Entonces, al estar las imágenes confinadas en una región del espacio, es decir, que tienen una extensión finita, la información de la fase es suficiente para reconstruirla (podemos ver la imagen a grandes rasgos).

La imagen de “Solo magnitud” es irreconocible y tiene severos problemas de rango dinámico. En cambio, la imagen de “Solo Fase” es reconocible, pero significativamente degradada en calidad. Si observamos nuevamente con mayor detenimiento la Fig. 5, podemos ver que la fase se distribuye casi por igual en todo el espectro de frecuencias mientras que el módulo se concentra en el centro del espectro y decae de manera exponencial al aumentar la frecuencia. [8]

Tomemos otro ejemplo para reafirmar lo anteriormente expuesto, en este caso, con una fotografía astronómica muy popular como lo es “moon.tif”. Aplicamos otra vez los procesamientos detallados en las Fig.4 y Fig.6, y exponemos los resultados en la Fig. 8. Vemos nuevamente el rol preponderante que tiene la fase.

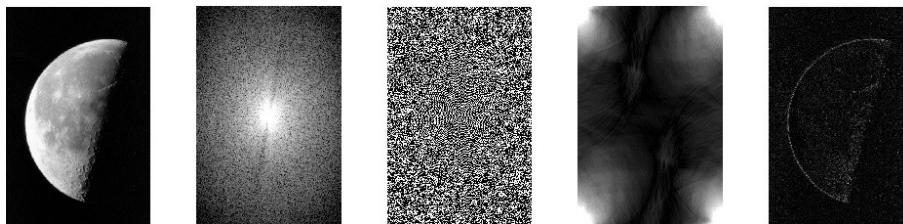


Fig. 8. De izquierda a derecha: imagen original “moon.tif”, “Módulo de la TDF”, “Fase de la TDF”, “Solo módulo” y “Sólo fase”. Todas ellas obtenidas a partir de la implementación de los procesamientos detallados en las Fig.4 y Fig. 6.

3.3 Experimento de Oppenheim

Un experimento que ilustra claramente la observación de que las señales “Sólo fase” capturan mayor inteligibilidad de la señal que las de “Sólo módulo”, es el que se conoce como “Experimento de Oppenheim” [9]. Este consiste en reproducir una imagen combinando el módulo de una imagen fuente con la fase de otra imagen fuente, y observar cuál información de las dos imágenes fuente predomina en la imagen resultante.

Este experimento ha sido llevado a cabo por numerosos autores y con distintos tipos de imágenes (paisajísticas, faciales, astronómicas, etc.), arribando todos ellos a las mismas conclusiones que las expuestas anteriormente.



Fig. 9. Experimento de Oppenheim. Esquina superior izquierda: Imagen “monkey.tif” (IM1). Esquina superior derecha: Imagen “lena.tif” (IM2). Esquina inferior izquierda: Imagen reconstruida utilizando el módulo de IM1 y la fase de IM2. Esquina inferior derecha: Imagen reconstruida utilizando el módulo de IM2 y la fase de IM1. [8]

4 Conclusiones y trabajos a futuro

En la representación de Fourier, el módulo y la fase tienden a desempeñar papeles diferentes y, en algunas situaciones, muchas de las características importantes de una señal se conservan si solo se mantiene la fase. Además, bajo una variedad de condiciones, como cuando una señal es de longitud finita, la información de la fase por sí sola es suficiente para reconstruir completamente una señal dentro de un factor de escala [9]. Esta observación acerca de la fase puede realizarse en varios contextos y aplicaciones diferentes, incluyendo tanto señales unidimensionales como bidimensionales y/o tridimensionales.

Los algoritmos de reconstrucción de imágenes “Solo Fase” se han explotado en determinadas áreas del procesamiento digital de imágenes. Por ejemplo; segmentación (dividir una imagen digital en varias partes u objetos), detección de bordes, procesos de eliminación de ruido, etc. Además, el estudio de la representación Módulo-Fase de la Transformada de Fourier en Tiempo Discreto es especialmente relevante en el diseño de filtros. Las citadas son líneas que se podrían desarrollar en una futura tarea de investigación.

Para finalizar, consideramos que, al vincular las materias básicas con las específicas de la ingeniería elegida, los estudiantes comprenden que todo saber se basa en un saber anterior, aunque sea el correspondiente al de una etapa que consideran ya superada, lo que contribuye a la idea de que cada asignatura no es un compartimiento estanco. Durante el desarrollo de la articulación de asignaturas planteado inicialmente, se estimula a que el alumno realice diversas actividades vinculadas a cada área, creando sus propios códigos para tal fin.

Creemos que la formación en la praxis se emprende descubriendo la formación sistémica del ingeniero, tratando de lograr una integración superadora de la visión parcial de cada una de las disciplinas, contribuyendo al desarrollo de las competencias de criterio y acción, a partir del conocimiento de las problemáticas de la ingeniería y de la tecnología.

Referencias

1. CONFEDI: Competencias en Ingeniería. *Web*. https://confedi.org.ar/download/documentos_confedi/Cuadernillo-de-Competencias-del-CONFEDI.pdf (2014). Accedido en septiembre 2015
2. Savoie, L.; Klimovsky, E.; Gaitán, M.: Aplicación de la Descomposición en Valores Singulares en la Compresión de Imágenes Digitales. *Libro de actas: XX Encuentro Nacional y XII Internacional de Educación Matemática en Carreras de Ingeniería*, pp. 61-68 (2017)
3. Cuevas Jimenez E.; Zaldivar Navarro D.; Rojas R.: *Computer Vision using MatLAB and the Toolbox of Image Processing*. Freie Universitat Berlin, pp. 7 (2005)
4. Gonzalez, Rafael C.; Woods, Richard E.: *Digital Image Processing*. Prentice Hall, pp. 150-154 (2001)
5. Avendaño, Santiago; Castillo, Gonzalo: *Detección y Reconocimiento de las Señas del Juego del Truco en Tiempo Real*. Tesis de Licenciatura. Facultad de Ciencias Exactas y Naturales. Universidad de Buenos Aires, pp. 7 (2012).
6. Aristizábal Ramírez, Diego Luis; Ramirez Martínez, Diego Alberto: *Conceptos Básicos del Procesamiento Digital de Imágenes Usando OrquideaJAI*. Escuela de Física, Universidad Nacional de Colombia - Sede Medellín, pp. 8-23 (2006)
7. Gonzalez, Rafael C.; Woods, Richard E.; Eddins, Steven L.: *Digital Image Processing using MatLab*. Pearson, pp. 112-115 (2003)
8. Skarbnik, Nikolay; Zeevi, Yehoshua Y.; Sagriv, Chen: The Importance of Phase in Image Processing. *Center for Communication and Information Technologies (CCIT) – Report #773* (2010)
9. Oppenheim, Alan V.; Lim, Jae S.; The Importance of Phase in Signals. *Proceedings of the IEEE*, vol. 69, no. 5, pp. 529-541 (1981)