PROCESADORES ÓPTICOS VIRTUALES DE ENCRIPTACIÓN USANDO LLAVES EN ARMÓNICOS CIRCULARES

CARLOS ALBERTO PINZÓN RUEDA

GRUPO ÓPTICA MODERNA
DEPARTAMENTO DE FÍSCA Y GEOLOGIA
FACULTAD DE CIENCIAS BÁSICAS
UNIVERSIDAD DE PAMPLONA
2016

PROCESADORES ÓPTICOS VIRTUALES DE ENCRIPTACIÓN USANDO LLAVES EN ARMÓNICOS CIRCULARES

CARLOS ALBERTO PINZÓN RUEDA

Director
JORGE ENRIQUE RUEDA PARADA
Doctor Ciencias Naturales Física

Trabajo de grado para optar al título de Físico

GRUPO ÓPTICA MODERNA
DEPARTAMENTO DE FÍSCA Y GEOLOGÍA
FACULTAD DE CIENCIAS BÁSICAS
UNIVERSIDAD DE PAMPLONA
2016

Nota de aceptación:

Firma del Director

Firma del jurado

Firma del jurado

Agradecimientos

A mi familia por su apoyo incondicional.

A mi director de tesis el Dr. Jorge Enrique Rueda por su paciencia, sus enseñanzas y orientación.

A mis amigos por toda la ayuda que me brindaron.

En general a todos aquellos que de forma directa e indirecta contribuyeron a alcanzar esta meta.

Tabla de Contenido

Lista de Figuras

Lista de Figuras Anexo

Lista de Tablas

Procesadores ópticos virtuales de encriptación usando llaves en armónicos circulares

Carlos A. Pinzón; Jorge-Enrique Rueda-P
Grupo Óptica Moderna, Universidad de Pamplona, Pamplona-Colombia

Resumen

Procesadores ópticos virtuales PEVLC y PEJTC para encriptar imágenes RGB fueron implementados y estudiados. La varianza del proceso de desencriptación con la rotación de la llave en coordenadas cartesianas y en componentes armónicas circulares (CHC) fue estudiada computacionalmente. Determinamos que la llave en coordenadas rectangulares permite una tolerancia de hasta ±1° de rotación de la llave. Con llaves en CHC encontramos una tolerancia de ±60° de rotación, sin utilizar máscara multiplicando la imagen de entrada. Varianza total fue encontrada cuando se multiplica la entrada con una máscara de amplitud de estructura periódica. También una novedosa estrategia de encriptación, que denominamos camuflaje de criptogramas fue presentada; desarrollamos un procesador óptico virtual para tal propósito.

Virtual optical encryption processors using keys in circular harmonics

Carlos A. Pinzón; Jorge-Enrique Rueda-P
Grupo Óptica Moderna, Universidad de Pamplona, Pamplona-Colombia

Abstract

Virtual optical processors PEVLC and PEJTC to encrypt RGB images were implemented and studied. The variance of the decryption process with the rotation of the key in Cartesian coordinates and in circular harmonic components (CHC) was studied computationally. We determined that the key in rectangular coordinates allows a tolerance of up to ± 1 ° of rotation of the key. With keys in CHC we find a tolerance of ± 60 ° of rotation, without using a mask multiplying the input image. Total variance was found when the input was multiplied with a periodic structure amplitude mask. Also a novel encryption strategy, which we call cryptogram camouflage was introduced; we developed a virtual optical processor for this purpose.

Capítulo I

1. INTRODUCCIÓN

A lo largo de los años el ser humano se ha visto en la necesidad de compartir información de manera segura en el dominio de lo no público; a partir de esta necesidad se han creado diferentes metodologías para ocultar o encriptar la información, de tal forma que solo el destinario tenga acceso a la información enviada. El término criptografía viene del griego κρύπτος (*criptos* = oculto), y γραφη (*grafé* = grafo o escritura); literalmente traduce "escritura oculta". La disciplina encargada de diseñar y construir los métodos de encriptación/desencriptación de información se conoce como criptografía, y a la información encriptada se le ha dado el nombre de criptograma. Los avances de la física y las matemáticas han permitido el desarrollo de diferentes técnicas o metodologías de encriptación cada vez de mayor robustez en cuanto a la seguridad de la información.

El origen de la criptografía es tan antigua como la escritura misma, impulsada principalmente por aspectos militares, religiosos o comerciales [1], los primeros intentos de la criptografía se halla en la antigua Egipto y los antiguos babilonios, donde la misma escritura era considerada código secreto [2, 3, 1]. El primer método de criptografía data del siglo V A.C., inventado por los éforos espartanos, conocida como *Escitela Lacedemonia*; en una vara con cierto diámetro se enrollaba en espiral una tira de cuero, de tal forma que se escribía el mensaje en columnas paralelas al eje de la vara, lo que hacía que al desenrollar la tira se leyera un mensaje sin coherencia; el mensaje solo se podía leer si se enrollaba en

una vara con el mismo diámetro de la vara inicial (este método se denomina transposición o permutación). Otro de los grandes referentes en la historia de la criptografía se originó en Roma, y fue utilizada por Julio Cesar en sus campañas de guerra. El método consiste en sustituir letras del mensaje. La *Escitela* y la encriptación de Cesar son los dos referentes más importantes de la criptografía en la antigüedad. En la edad media la criptografía sufre un estancamiento [2, 3]; fue hasta siglo XV que León Battista Alberti, considerado el padre de la criptografía, desarrolló el primer sistema poli alfabético conocido como disco de Alberti.

Es bien conocido que una buena parte de los avances tecnológicos han surgido a partir de las necesidades militares para la guerra; dos ejemplos cercanos son la primera y segunda guerra mundial, en donde los sistemas criptográficos tuvieron importantes avances, por ejemplo, en la segunda guerra mundial se inventó la máquina "Enigma", construida por Arthur Scherbius y Richard Ritter en el año de 1918 [2, 4]. En la **Tabla 1** se muestra una clasificación de las estrategias clásicas y modernas de la criptografía.

Tabla 1. Clasificación de la criptografía clásica y moderna

	Técnicas criptográficas	Ejemplos
Criptografía Clásica	**Sustitución** Se sustituye las letras del texto original por otras que se encuentren ubicadas en un determinado número de espacios del alfabeto.	Cifrado de Cesar, disco de Alberti, cifrado de Vigenére, etc.
	Transposición Se reorganiza las letras del mensaje de tal forma que sea ilegible y por medio de una clave (llave) se hace legible el mensaje.	Escitela, carril Fence, etc.
Criptografía Moderna	**Simétrica** Utiliza una única clave (llave) para encriptar y desencriptar la información.	Enigma, DES, IDEA, AES, etc.
	Asimétrica Utiliza una llave pública y una privada; la pública para encriptar y la privada para desencriptar la información.	RSA, PGP, DSA, criptografía de curva elíptica, etc.

La era digital y las nuevas tecnologías de la informática dieron origen a un nuevo escenario de la criptografía; desde enviar un correo electrónico, hasta una llamada telefónica, dichos procesos se rigen por sistemas criptográficos. Sin embargo, a pesar de los grandes avances tecnológicos los sistemas criptográficos aun no son 100% seguros, quedando así abierta esta brecha de conocimiento para nuevas investigaciones en criptografía clásica y más recientemente usando los principios de la mecánica cuántica [5, 6].

Una de las líneas de la criptografía moderna surge en las técnicas de procesamiento óptico, siendo esta la línea la base de los resultados de este trabajo de investigación. La técnica de encriptación óptica resulta de interés por su alta capacidad y velocidad de procesamiento en paralelo de la información [3, 7, 8]. Entre otras ventajas de la encriptación óptica, está la posibilidad de múltiples grados de libertad a la hora de encriptar la información, por ejemplo, se puede encriptar en términos de la fase, de la longitud de onda o de la polarización de la luz, el uso de llaves en cascada, multiplexado, uso de diferentes tipos de operadores, tipos de transformación, entre otras estrategias, y con ello el nivel de seguridad ha sido mejorado de manera importante [9, 10, 11].

Se espera que en un futuro tales ventajas de la óptica permitan obtener procesadores de encriptación ópticos tan robustos como para que funcionen con el 100% de seguridad. En la **Tabla 2** se relacionan algunos aportes en criptografía óptica, los cuales consideramos de mayor relevancia, y los clasificamos según utilicen alguna de las arquitecturas de correlación *Vander Lugt Correlator* (VLC) [12] ó *Join Transform Correlator* (JTC) [13]; estas dos arquitecturas son la base de la mayoría de los arreglos de encriptación óptica.

Tabla 2. Publicaciones de criptografía óptica basadas en las arquitecturas VLC y JTC

Arquitectura VLC de Encriptación	***Arquitectura JTC de Encriptación***
Encriptación usando doble mascara de fase (DRPE). [7]	Uso de multiplexado angular en materiales fotorrefractivos. [14]
Encriptación usando una longitud de onda específica. [15]	Encriptación de objetos 3D usando técnicas de holografía digital. [16]
Encriptación completamente en fase. [17, 18]	Optimización de la llave de encriptación, aumentando la relación de señal/ruido. [19]
Encriptación usando técnicas de holografía digital y la transformada rápida de Fourier (FFT). [20]	Encriptación mediante interferometría de corrimiento de fase y la transformada de Fresnel. [21]
Encriptación en cascada usando la transformada de Fourier fraccional. [22]	Creación de la llave por medio de moduladores espaciales de luz de cristal líquido. [23]
Multiplexado de imágenes encriptadas. [24]	Encriptación usando múltiples llaves desfasadas en cuatro canales. [25]
Encriptación usando la transformada coseno fraccional discreta de múltiples órdenes. [26]	Encriptación de múltiples imágenes usando llaves de encriptación fractal. [27, 28]
Uso de la transformada de Hartley y la teoría del caos en la creación de las llaves de encriptación. [29]	Encriptación de múltiples videos usando un interferómetro Mach-Zehnder. [30]
Encriptación usando la transformada radial de Hilbert. [31]	Encriptación de múltiples imágenes usando luz polarizada en el dominio de la transformada de Fresnel. [32]
Encriptación de imagen a color por medio de multiplexado en longitud de onda, y usando la transformada fraccional de Hartley. [33]	Encriptación de imágenes a color usando multiplexado espacial y operaciones de truncamiento de fase. [34]
Encriptación de múltiples imágenes usando interferometría y DRPE. [35]	Encriptación y compresión de la información mediante modulación de polarización aleatoria. [36]
Encriptación de objetos 3D usando un sistema multi-espectral de imagen integrada computacionalmente. [37]	
Encriptación de imágenes/videos usando DRPE y técnicas de holografía digital en el dominio de Fresnel. [38]	
Encriptación de objetos de fase usando lentes de Fresnel de perfil vórtice del diablo (DVFL). [39]	
Encriptación usando llaves en armónicos circulares. [40]	
Encriptación usando máscaras de fase determinísticas [41]	

Los objetivos de este trabajo de investigación fueron orientados al problema de la varianza a la rotación de la llave en procesadores óptico virtuales de encriptación, PEVLC -Procesador de Encriptación VLC- y PEJTC -Procesador de Encriptación JTC-. La solución del problema se abordó desde dos perspectivas, la primera usando llaves con distribución de fase en coordenadas cartesianas, y en la segunda usando llaves de solo fase en componentes armónicas circulares (CHC: *Circular harmonic components*).

Para mejor entendimiento y coherencia en la presentación de los resultados del trabajo, en la estructura de este informe se ha incluido un capítulo (Capítulo II) donde se introducen aspectos físico-matemáticos de la difracción y su relación con la transformada de Fourier óptica; la modelación analítica y computacional de los procesadores ópticos de encriptación PEVLC y PEJTC, usando llaves con distribución de fase en coordenadas cartesianas; y finalmente resultados y discusión referida a la varianza a la rotación de llave en el proceso de desencriptación.

En el Capítulo III se hace una presentación del proceso de descomposición en armónicos circulares, mostrando resultados de como computacionalmente se puede transformar una imagen en coordenadas cartesianas a una imagen en CHC; mostramos resultados del cálculo de llaves de encriptación en CHC, partiendo de una distribución de fase en coordenadas cartesianas.

El Capítulo IV contiene los resultados obtenidos del estudio de la varianza a la rotación usando llaves en CHC. Además presentamos resultados inéditos de una nueva estrategia de encriptación que denominamos camuflaje de criptogramas, usando órdenes enteros y fraccional de la descomposición CHC de la llave; este resultado motivó el desarrollo de una herramienta de encriptación óptica virtual; cuyo manual técnico y de usuario se ha incluido como Anexo I.

Capítulo II

2. TRANSFORMADA DE FOURIER Y PROCESADORES PEVLC Y PEJTC

2.1. TRANSFORMADA DE FOURIER

La transformada de Fourier llamada así en honor a Joseph Fourier, fue desarrollada para funciones periódicas y no periódicas [42]; esta transformada convierte una función del dominio espacial o temporal, al dominio de las frecuencias y viceversa.

Definición: sea $\boldsymbol{f(x,y)}$ continua y de cuadrado integrable. Entonces su transformada de Fourier $\boldsymbol{\Im\{f(x,y)\}}$ en coordenadas cartesianas es:

$$\boldsymbol{F(u,v) = \int\int_{-\infty}^{\infty} f(x,y) e^{-i2\pi(xu+yv)} dx\, dy,} \qquad \textbf{2.1}$$

y su transformada inversa $\boldsymbol{\Im^{-1}\{F(u,v)\}}$ es:

$$\boldsymbol{f(x,y) = \int\int_{-\infty}^{\infty} F(u,v) e^{i2\pi(xu+yv)} du\, dv,} \qquad \textbf{2.2}$$

Donde $\boldsymbol{(u,v)}$ son coordenadas de frecuencias espaciales relativas a las coordenadas $\boldsymbol{(x,y)}$. En este trabajo, la transformada de Fourier es la operación matemática mediante la cual se explica y modelan los procesadores de encriptación PEVLC y PEJTC.

De otra parte, es conocido que la transformada de Fourier está ligada al fenómeno de difracción de la luz y su estudio se conoce como la teoría escalar de la difracción [42]. Haciendo una breve reseña histórica sobre este fenómeno natural, se sabe que el primer reporte científico es de autoría de Francesco Grimaldi con el título "*físico mathesis de Lumine, coloribus, et iride*" [43]; Grimaldi acuñó la palabra

"*diffractio*". Entre los aportes a la interpretación del fenómeno de difracción, Arnold J. Sommerfeld definió el fenómeno como: "Toda desviación de los rayos luminosos de su trayectoria rectilínea, que no puede explicarse ni por reflexión ni por refracción". Cristian Huygens, expone por primera vez la teoría ondulatoria en 1678, y explica su concepción intuitiva del fenómeno de propagación: "Si cada punto de la superficie de una onda de una perturbación luminosa es considerada como la fuente de una nueva perturbación esférica secundaria, se podría determinar la superficie de onda en cualquier instante posterior mediante la construcción de la envolvente de tales ondas secundarias". Agustín Jean Fresnel, en 1818 desarrolló las bases matemáticas, basándose en el principio de Huygens, y refuerza el principio mediante la interferencia de Young. Gustav Kirchhoff (1822), plantea su formulación al problema de difracción basándola en un teorema integral que expresa la solución de la ecuación de onda homogénea como un problema de valores de frontera de la luz sobre la superficie de un obstáculo. Poincare (1892) y Sommerfeld (1894) mostraron que las ecuaciones matemáticas de Kirchhoff eran falsas, y que el llamado principio de Huygens-Fresnel es válido en primera aproximación. Sommerfeld (1894) elimina una de las ecuaciones de Kirchhoff concerniente a las condiciones de frontera, haciendo uso de las funciones de Green; esto conduce a lo que se conoce como teoría de difracción Raileigh-Sommerfeld. Kottler (1965); resuelve las contradicciones de la teoría de Kirchhoff, reinterpretando el problema de "saltus" ó discontinuidades de una función. De todos estos estudios sobre la difracción, que aún siguen siendo objeto de estudio, la pregunta es cuál es la relación con la transformada de Fourier. En sumario, en este tratamiento escalar del problema se han establecido tres dominios de aproximación o soluciones que se muestra en la **Figura 2.1**.

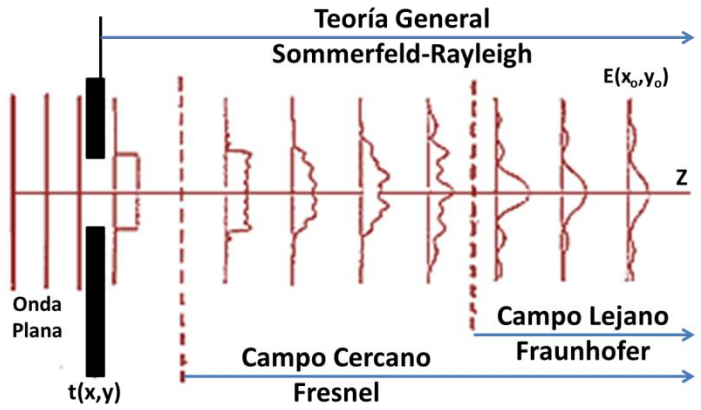


Figura 2.1. Aproximaciones de la teoría escalar de la difracción.

La aproximación de campo cercano, también conocida como aproximación de Fresnel, se define por la **Ec.2.3**:

$$\boldsymbol{E(x_0,y_0,z) = \frac{e^{ikz}}{i\lambda z} \cdot e^{-\frac{ik}{2z}(x_0^2+y_0^2)} \int\int_{-\infty}^{\infty} E(x,y) e^{\frac{ik}{2z}(x^2+y^2)} e^{-\frac{ik}{z}(x_0x+y_0y)} dx\, dy} \qquad \textbf{2.3}$$

Donde $\boldsymbol{E(x_0,y_0)}$ es el campo observado sobre plano $(\boldsymbol{x_0,y_0})$ ubicado a la distancia z del objeto difractor de transmitancia $\boldsymbol{t(x,y)}$; $\boldsymbol{E(x,y)}$ es el producto del frente de onda por la transmitancia $\boldsymbol{t(x,y)}$ del objeto ubicado en $\boldsymbol{z=0}$; $\boldsymbol{k}$ es el número de onda y $\boldsymbol{\lambda}$ la longitud de onda de la fuente de iluminación. Podemos hacer el cambio de variable $\boldsymbol{u=\frac{x_0}{\lambda z}}$ y $\boldsymbol{v=\frac{y_0}{\lambda z}}$, donde $\boldsymbol{u}$ y $\boldsymbol{v}$ son frecuencias espaciales en el plano $(\boldsymbol{x_0,y_0})$. Si consideramos un plano de observación tal que $\boldsymbol{z \gg}$ que las dimensiones del objeto difractor, entonces el factor de fase cuadrático $\boldsymbol{e^{\frac{ik}{2z}(x^2+y^2)} \cong 1}$ (aproximación denominada de Fraunhofer), así la **Ec.2.3** toma la forma:

$$\boldsymbol{E(u,v) = \frac{e^{ikz}}{i\lambda z} \cdot e^{i\pi(u^2+v^2)} \int\int_{-\infty}^{\infty} E(x,y) e^{i2\pi(xu+yv)} dx\, dy,} \qquad \textbf{2.4}$$

En la **Ec.2.4** el término integral es la transformada de Fourier del campo del plano del objeto difractor. En conclusión, la **Ec.2.4** es la difracción de Fraunhofer,

observada sobre el plano (x_0, y_0); esta es una transformada de Fourier no exacta del objeto difractor, o mejor podemos decir que es una transformada de óptica de Fourier, donde el plano de observación correcto es de forma paraboloide. También es posible obtener una transformada de Fourier usando un procesador óptico 2f como el de la **Figura 2.2.**

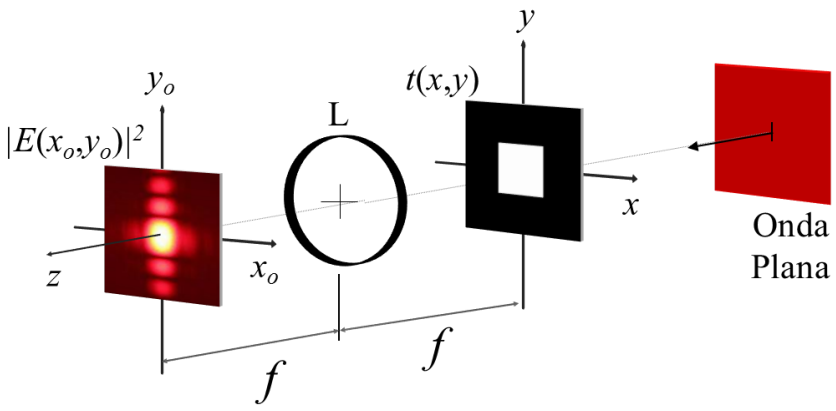


Figura 2.2. Arreglo de difracción 2f. f es la longitud focal de la lente convergente L. $t(x,y)$ es la transmitancia del objeto difractor. $E(x_0,y_0)$ es el campo difractado ó espectro de Fourier de $t(x,y)$.

Si $z = f$, entonces la **Ec.2.4** toma la forma:

$$E(u,v,f) = A \cdot \int\int_{-\infty}^{\infty} t(x,y) e^{-i2\pi(xu+yv)} dx\, dy, \quad \therefore A = \frac{e^{i\left(kf-\frac{\pi}{2}\right)}}{\lambda f} \qquad \textbf{2.5}$$

En este caso la repartición de intensidad sobre el plano focal de la lente, dada por la **Ec.2.6**, es aproximadamente igual al espectro de potencia de Fourier de la transmitancia del objeto difractor, excepto por el factor constante $\frac{1}{(\lambda f)^2}$.

$$I(u,v,f) \approx \left| \int\int_{-\infty}^{\infty} t(x,y) e^{-i2\pi(xu+yv)} dx\, dy \right|^2 \qquad \textbf{2.6}$$

Todo este marco conceptual y matemático sirve para recordar que mediante el fenómeno de difracción se pueden calcular transformadas de Fourier. Esta

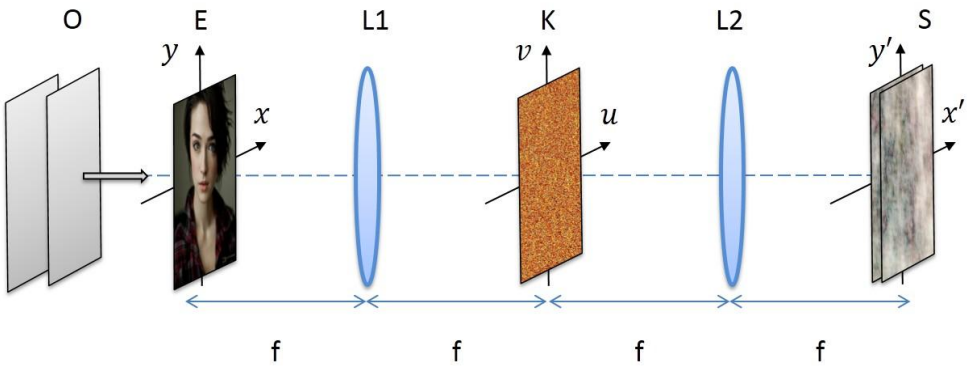
transformada es parte esencial de este trabajo de investigación; la razón se debe a que los procesadores de encriptación ópticos tanto el PEVLC como el PEJTC son la combinación de procesadores 2f como el de la **Figura 2.2**. La combinación de estos procesadores 2f, nos permite obtener operaciones de convolución, entre la información que se desea encriptar, con un filtro de solo fase (de valores aleatorios denominado llave de encriptación/desencriptación). Mediante la operación de convolución el resultado esperado es una imagen oculta, que en adelante llamaremos imagen encriptada; el problema inverso es la desencriptación, cuyo resultado es la imagen desencriptada. En un trabajo previo publicado en 2015 [40], se demostró que cuando la llave está definida en coordenadas cartesianas el proceso de desencriptación puede generar dificultades de operación debido a que es variante con la rotación de la llave (esto tiene sentido en procesadores de encriptación totalmente ópticos). Haciendo una modelación computacional se encontró que no son totalmente variantes, que existe una tolerancia a la rotación de la llave de aproximadamente $\pm 0.1°$. Estos resultados previos, dieron origen al presente trabajo de investigación, donde hemos profundizado el problema de manera computacional. Trabajamos con el algoritmo que calcula la transformada Rápida de Fourier Discreta desarrollado por MathWorks, y donde no consideramos el factor *A* en los desarrollos computacionales implementados, teniendo en cuenta que este factor es una constante que en términos generales no afecta conceptualmente los resultados. La transformada de Fourier Discreta se define mediante la siguiente expresión:

$$\boldsymbol{F(u,v) = \frac{1}{N \cdot M} \sum_{X=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) e^{-i2\pi\left(u\frac{x}{M}+v\frac{y}{N}\right)}}, \qquad \textbf{2.7}$$

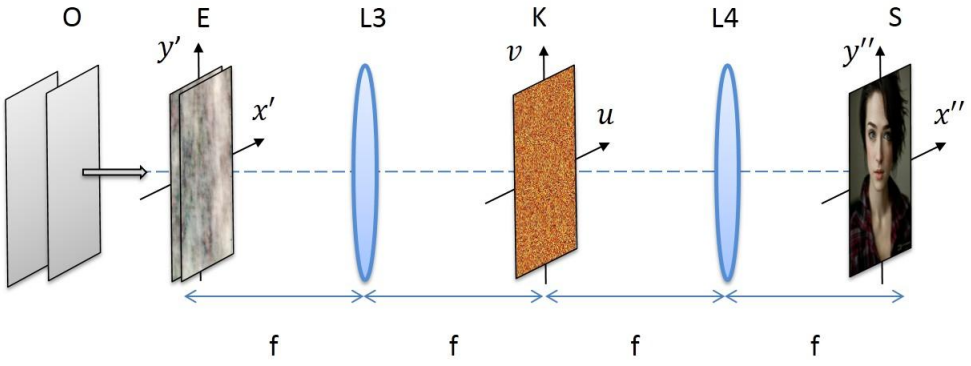
Donde $\boldsymbol{u = 0,\ldots,(M-1)}$; $\boldsymbol{v = 0,\ldots,(N-1)}$, siendo $N \times M$ el tamaño de la imagen $\boldsymbol{f(x,y)}$.

2.2. ANALISIS FÍSICO-MATEMÁTICO DEL PROCESADOR PEVLC

En el capítulo I se dio una reseña del desarrollo histórico de la encriptación, en particular se hizo énfasis en aquellos esquema que usan los principios de la óptica clásica. En este apartado se presenta el estudio analítico y computacional de las dos arquitecturas ópticas de mayor uso en los procesadores de encriptación óptico; arquitectura tipo Vander Lugt Correlator (VLC) y la tipo Joint Transform Correlator (JTC). Centramos este estudio computacional en el problema de la varianza del proceso de desencriptación con la rotación de la llave.



(a). Encriptación



(b). Desencriptación

Figura 2.3. Esquemas de encriptación-desencriptación. O: onda plana; E es el plano de entrada; L1, L2, L3, L4 son lentes convergentes; siendo f la distancia focal de cada lente. K es la llave; S es el plano de salida.

Es conocido que el VLC permite medir el grado de similitud entre dos funciones, en donde una función se denomina escena de entrada y la otra se denomina la escena de referencia u objetivo [44]. El procesador PEVLC tiene la misma estructura de construcción del VLC, pero su diferencia está en el modo de operación, y en que en el plano espectral se ubica un filtro de solo fase, que en coordenadas cartesianas tiene la forma $\boldsymbol{K_{e,d}(u,v) = e^{i\Phi(u,v)}}$, que en adelante nos referiremos a la llave de encriptación $\boldsymbol{K_e}$ o desencriptación $\boldsymbol{K_d}$. El parámetro $\boldsymbol{\Phi(u,v)}$ representa una distribución de fase de valores aleatorios, que deben estar distribuidos entre $[-\boldsymbol{\pi},\boldsymbol{\pi}]$ para que la operación de encriptación genere una imagen encriptada. La **Figura 2.3(a)** muestra el esquema del procesador de encriptación, y la **Figura 2.3(b)** representa el procesador de desencriptación.

En la **Figura 2.3(a)**, en el plano de entrada $(\boldsymbol{x,y})$ se tiene la escena a encriptar $\boldsymbol{f(x,y)}$, este plano se ilumina con un frente de onda plano, monocromático. A través de la lente L1 se obtiene, en su plano focal, la transformada de Fourier $\boldsymbol{F(u,v)}$ del plano de entrada; esta transformada se multiplica por la llave de encriptación $\boldsymbol{K_e(u,v)}$ que está ubicada justo sobre el plano focal de la lente L1, así se produce la multiplicación $\mathbf{F(u,v)} \cdot \boldsymbol{K_e(u,v)}$. Mediante la lente L2 se obtiene, en el plano de salida $(\boldsymbol{x',y'})$, la convolución dada por la siguiente relación:

$$\boldsymbol{f_e(x',y') = f(x',y') * k_e(x',y')}, \qquad \textbf{2.8}$$

Donde la **Ec.2.8** representa la imagen encriptada $\boldsymbol{f_e(x',y')}$, que es el resultado de la convolución entre la escena de entrada y la transformada de la llave $\boldsymbol{k_e(x',y')}$. La transformada de la llave es también un ruido de solo fase de distribución aleatoria.

El esquema de la **Figura 2.3(b)** es la representación del arreglo de desencriptación. El plano de entrada contiene la imagen $\boldsymbol{f_e^*(x',y')}$ que representa el conjugado de la imagen encriptada. Mediante L3 se obtiene en su plano focal la

transformada de Fourier $\boldsymbol{F_e^*(-u,-v)}$ del conjugado de la imagen encriptada; en este mismo plano, $\boldsymbol{F_e^*(-u,-v)}$ se multiplica por la llave de desencriptación $\boldsymbol{k_d(u,v)}$, esto es:

$$\boldsymbol{\Im\{f_e^*(x',y')\}\cdot K_d(u,v) = [F(u,v)\cdot K_e(u,v)]^*\cdot K_d(u,v)} \qquad \textbf{2.9}$$

Finalmente, a través de L4, y en su plano focal, se produce la imagen desencriptada según la **Ec.2.10**,

$$\boldsymbol{f_d(x'',y'') = f(x'',y'') * \{k_e{}^*(x'',y'')\odot k_d(x'',y'')\}.} \qquad \textbf{2.10}$$

Donde el producto de correlación $\boldsymbol{k_e{}^*(x'',y'')\odot k_d(x'',y'')}$ será igual a un pico dirac $\boldsymbol{\delta(x'',y'')}$ siempre que se cumpla que $\boldsymbol{K_e(u,v) = K_d(u,v)}$, entonces solo así la imagen será desencriptada sin distorsión. En caso contrario este producto de correlación genera un ruido de solo fase y con valores aleatorios que producen distorsión en el plano de salida debido a la convolución del mismo con $\boldsymbol{f(x'',y'')}$, en otras palabras, no hay desencriptación.

2.2.1. PEVLC usando llaves en coordenadas cartesianas: análisis de la varianza de la desencriptación con la rotación de la llave.

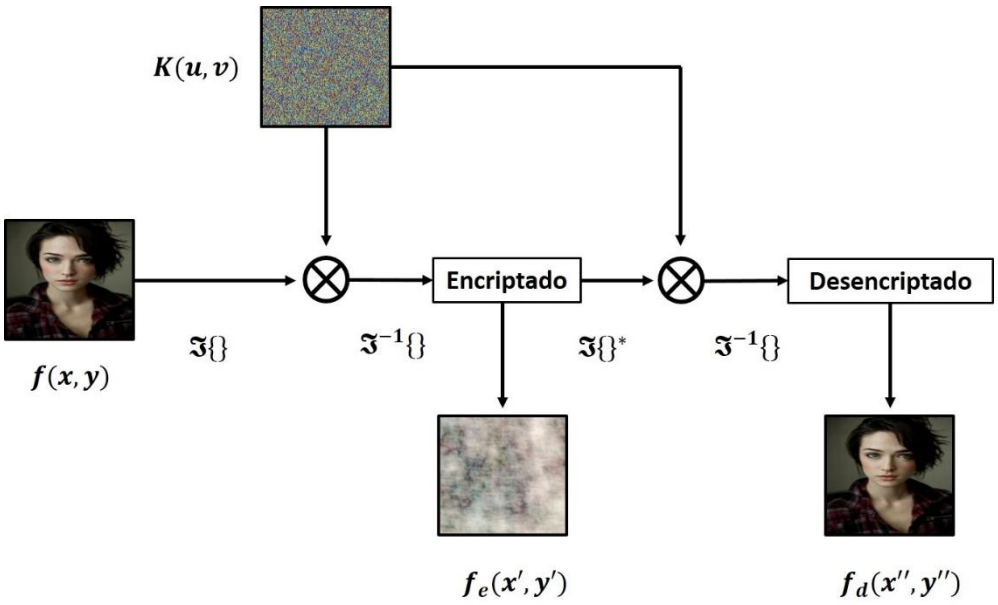


Figura 2.4. Diagrama de flujo del proceso de encriptación/desencriptación de un PEVLC; $\otimes$ operador multiplicación aritmética; $\mathfrak{I}\{\}$ operador transformada de Fourier.

En este apartado, discutimos los resultados de la modelación computacional de un PEVLC usando lenguaje MATLAB. La base de esta implementación es el modelo analítico hecho en el apartado anterior. En la **Figura 2.4** se muestra la lógica implementada en la simulación computacional.

Siguiendo este diagrama lógico del PEVLC se hizo el desarrollo computacional del sistema de encriptación/desencriptación para imágenes RGB de tamaño 400x400 pixeles (**Figura 2.6(a)**), teniendo en cuenta que el algoritmo se desarrolló con tres canales para que funcione cualquier tipo de imagen. Se calcularon llaves de fase aleatoria -ruido blanco- $\boldsymbol{K(u,v) = e^{i\phi(u,v)}}$ (ver imagen de la fase en la **Figura**

2.6(b)), con valores de fase entre $[-\pi,\pi]$. La **Figura 2.5** es el histograma típico de esta distribución de fase de las llaves calculadas.

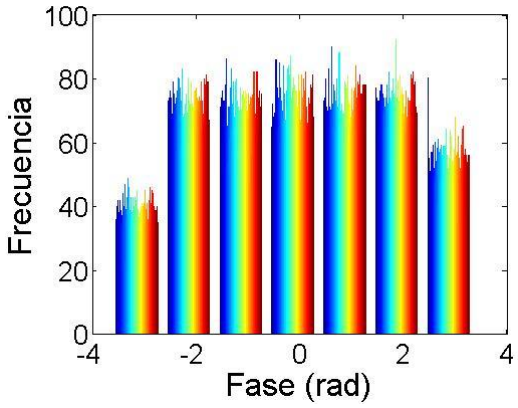
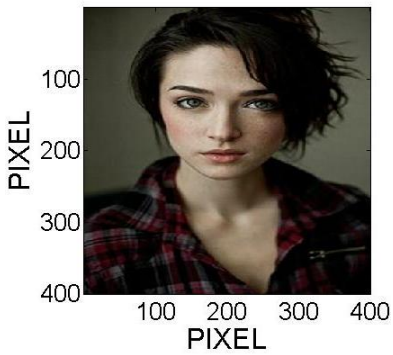
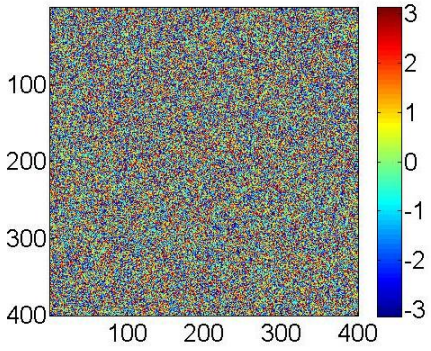


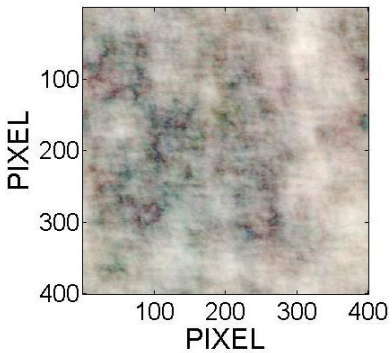
Figura 2.5. Histograma típico de la distribución de fase de las llaves calculadas.



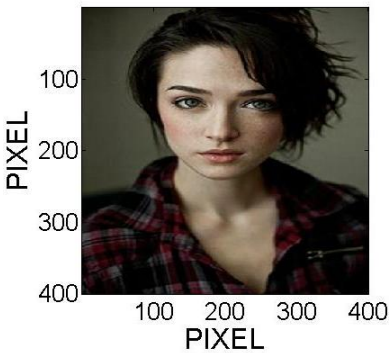
(a). Imagen de entrada



(b). Fase de la llave



(c). Amplitud de la imagen encriptada



(d). Imagen desencriptada

Figura 2.6. Resultado computacional utilizando un PEVLC y llave en coordenadas cartesianas.

La **Figura 2.6** contiene un resultado de una simulación en donde la llave no fue rotada. La **Figura 2.6(c)** es el resultado de la imagen encriptada y la **Figura 2.6(d)** es la imagen desencriptada. Subjetivamente podemos apreciar que la imagen desencriptada es igual a la imagen de entrada. Sin embargo, comprobamos cuantitativamente que la calidad de la imagen desencriptada no es igual a la entrada, hecho que nos permite inferir que persiste un cierto porcentaje de ruido en la imagen desencriptada, no perceptible por el ojo humano. Esta medición de calidad de la imagen se calculó mediante la relación PSNR (Peak *Signal Noise Ratio*) [45, 46] dada por la siguiente expresión:

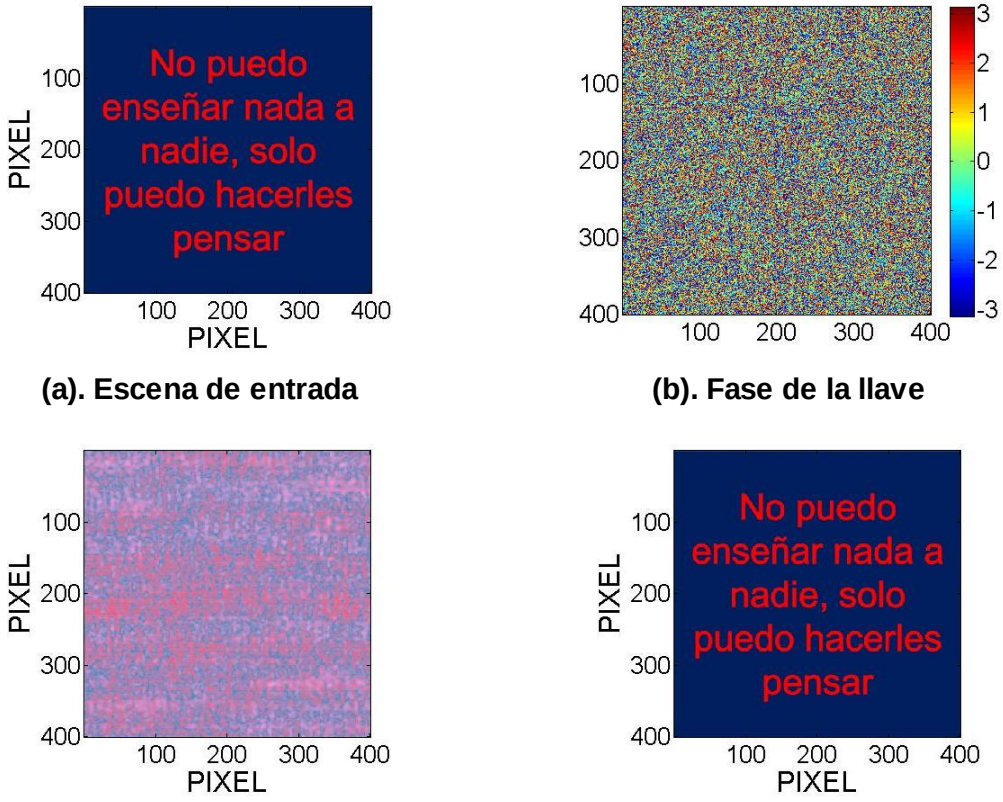
$$\boldsymbol{PSNR = 10 \cdot \log_{10}\left(\frac{[max\{f(x,y)\}]^2}{MSE}\right)} \qquad \textbf{2.11}$$

$$\boldsymbol{MSE = \frac{1}{MN}\sum_{x=1}^{M}\sum_{y=1}^{N}[f(x,y) - f_d(x,y)]^2,} \qquad \textbf{2.12}$$

Donde MSE (*Mean Square Error*) es el error cuadrático medio, $\mathbf{max}\{\mathbf{f}(\mathbf{x},\mathbf{y})\}$ es el valor máximo en la imagen a encriptar, $\boldsymbol{f_d(x,y)}$ es la imagen desencriptada, N es número de filas y M el de columnas de $\boldsymbol{f(x,y)}$ o $\boldsymbol{f_d(x,y)}$. Así por ejemplo, para el resultado de la **Figura 2.6**, la imagen desencriptada tiene una relación $PSNR = 52.46dB$.

Otro resultado se presenta en la **Figura 2.7**, para una escena de texto. El resultado nos permite concluir que tanto el nivel de encriptación, así como la calidad de la imagen desencriptada no es un asunto determinístico; esto implica que cada escena a encriptar requiere hacer ensayos particulares donde se verifique el nivel de encriptación. Obsérvese que en este caso del texto, aun teniendo una distribución de la llave estadísticamente buena, el nivel de encriptación es parcialmente bueno, en la medida que en la amplitud de la imagen encriptada un

experto podría inferir que existe rastro del texto. Más adelante en otros resultados, demostramos que se puede mejorar el nivel de encriptación para este tipo de objetos, multiplicando el plano de entrada por una máscara de solo fase, de valores randómicos o cualquier otro tipo de distribución no determinística, por ejemplo tipo Speckle. Con respecto a la calidad de la imagen desencriptada del texto, encontramos un $PSNR = 315.9dB$. Comparando esta medida con el PSNR del objeto rostro del resultado anterior, en el texto queda un residuo de ruido 6 veces menor en la imagen desencriptada del rosto, esto se debe a que la imagen texto es una imagen binaria.



(a). Escena de entrada **(b). Fase de la llave**

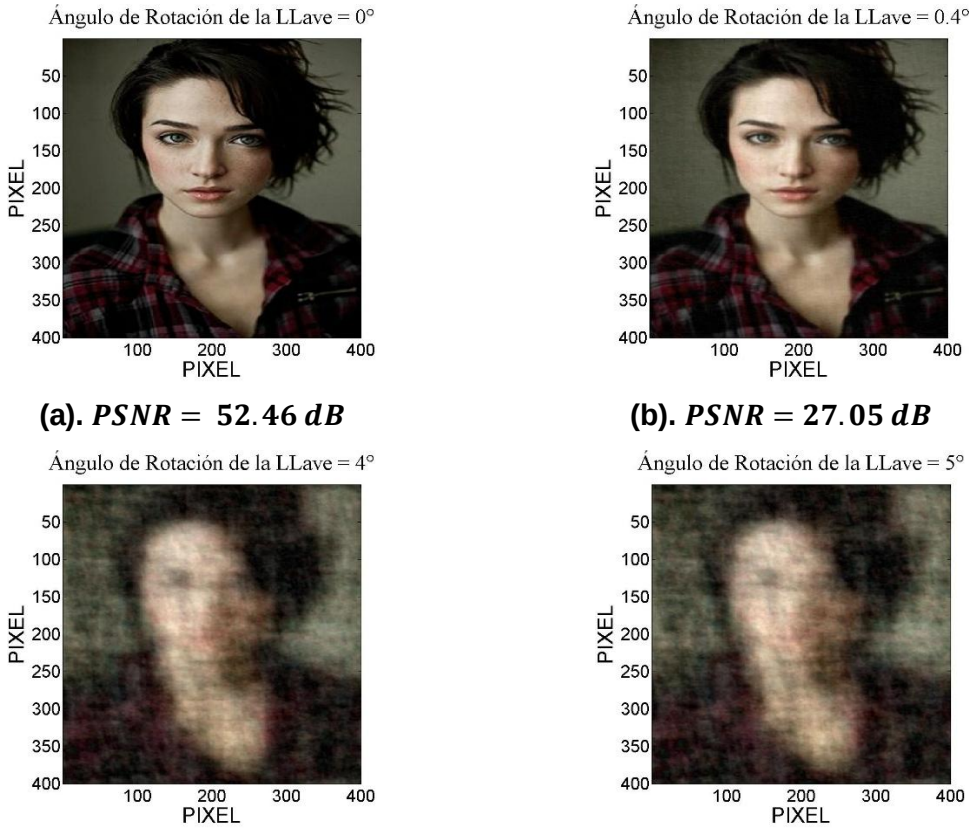
(c). Amplitud de la imagen encriptada **(d). Imagen desencriptada**

Figura 2.7. Resultado computacional utilizando un PEVLC y llave en coordenadas cartesianas.

Las **Figura 2.8-Figura 2.10** son algunos resultados del estudio de la varianza del proceso de desencriptación cuando se rota la llave durante este proceso. Dividimos estos resultados en dos casos:

CASO I. VARIANZA DE LA DESENCRIPTACIÓN CON LA ROTACIÓN DE LA LLAVE EN COORDENADAS CARTESIANAS, SIN MÁSCARA MULTIPLICANDO LA ESCENA DE ENTRADA.

Una vez encriptada la imagen se procedió a desencriptar girando la llave a paso de 0.1° y hasta 360°. La **Figura 2.8** contiene solo cuatro resultados intermedios y que corresponde a rotaciones de la llave de 0°,0.4°,4° y 5°.



(a). $PSNR = 52.46\ dB$ **(b).** $PSNR = 27.05\ dB$

(c). $PSNR = 19.84\ dB$ **(d).** $PSNR = 18.52\ dB$

Figura 2.8. Imágenes desencriptadas con rotación de la llave, sin utilizar máscara de fase multiplicando la escena de entrada.

Analizando subjetivamente estos resultados de la **Figura 2.8**, podemos establecer que para este caso la tolerancia a la rotación de la llave es baja, no superior a ±1°. En 4° y 5° se observa una distorsión que cualificamos como desencriptación

media o parcial. Por encima de 5° la distorsión es alta y por ende no hay desencriptación de la imagen. En la **Figura 2.9** se muestra el comportamiento de la PSNR en función de ángulo de rotación de la llave.

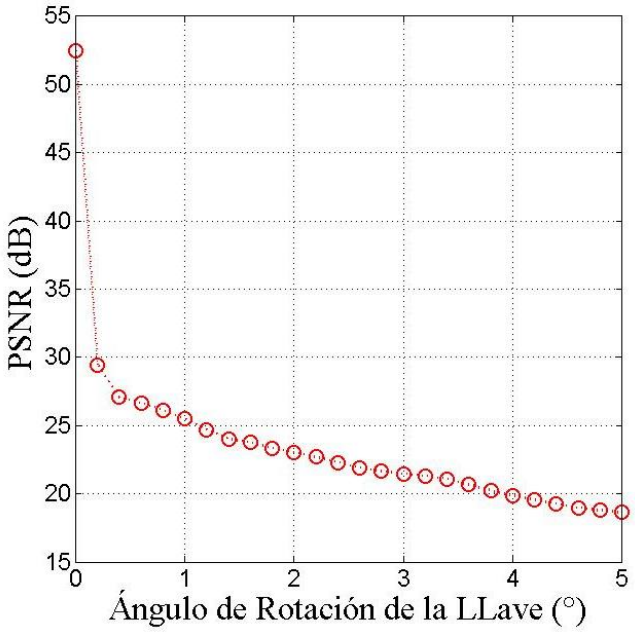
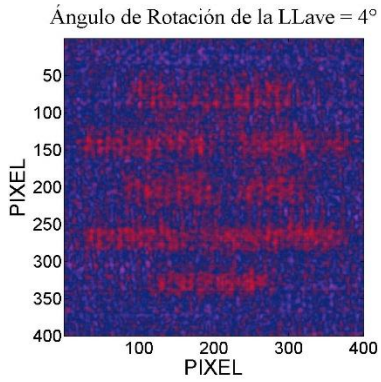


Figura 2.9. Correlación entre la PSNR de imagen desencriptada y el ángulo de rotación de la llave $K_d(u,v)$ caso, objeto rostro, sin máscara multiplicando la escena de entrada.

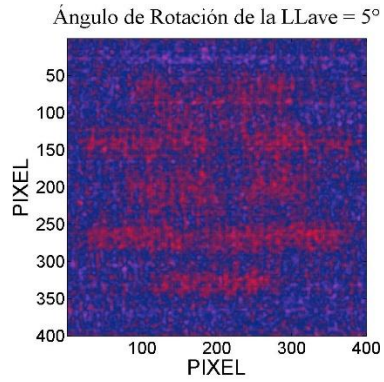
El mismo estudio computacional de rotación de la llave en la desencriptación se realizó para el objeto texto binario. La **Figura 2.10** muestra estos resultados, en los que se puede apreciar que el resultado es similar al caso del objeto rostro; la diferencia principal es el PSNR de cada imagen desencriptada, como se puede apreciar en la **Figura 2.11**, donde se muestra la correlación entre el PSNR de cada imagen desencriptada y el ángulo de rotación de la llave. La tolerancia a la rotación es similar al caso de la escena rostro.

(a). $PSNR = 315.9\ dB$

(b). $PSNR = 25.75\ dB$



(c). $PSNR = 14.12\ dB$



(d). $PSNR = 13.26\ dB$

Figura 2.10. Imágenes desencriptadas con rotación de la llave, caso objeto texto y sin utilizar máscara de fase multiplicando la entrada.

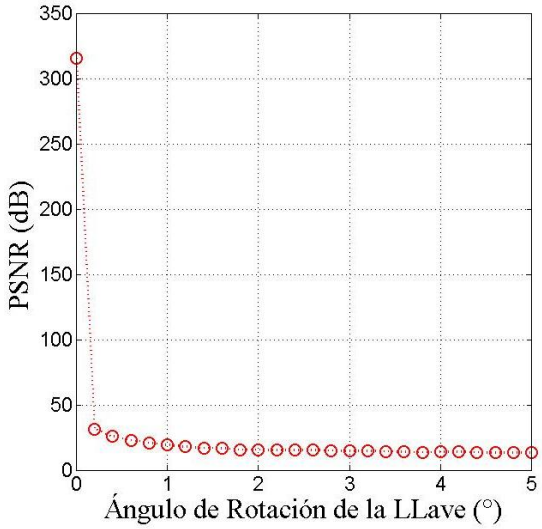
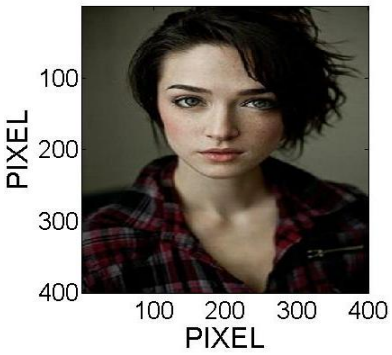


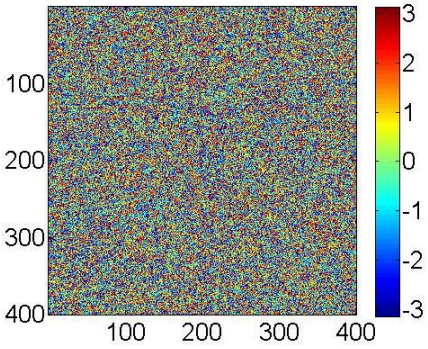
Figura 2.11. Correlación entre la PSNR de imagen desencriptada y el ángulo de rotación de la llave $K_d(u,v)$ caso objeto texto, sin máscara multiplicando la escena de entrada.

CASO 2. VARIANZA DE LA DESENCRIPTACIÓN CON LA ROTACIÓN DE LA LLAVE EN COORDENADAS CARTESIANAS, CON MÁSCARA MULTIPLICANDO LA ENTRADA.

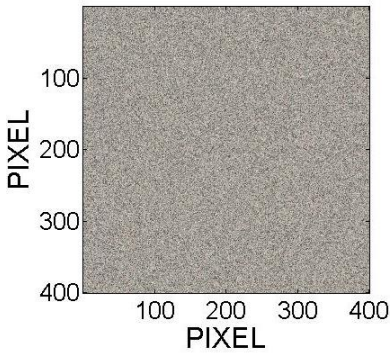
Como se puede apreciar en los resultados que muestra la **Figura 2.12**, particularmente en el resultado de la imagen encriptada, el nivel de encriptación es alto con respecto al caso anterior donde no se utilizó máscara multiplicando la entrada; sin embargo, el $PSNR = 52.46dB$ en la imagen desencriptada es el mismo del caso sin máscara, pero sin rotar la llave. Un efecto similar se encontró para el test con otros objetos, entre ellos, como la escena texto que se muestra en la **Figura 2.13**.



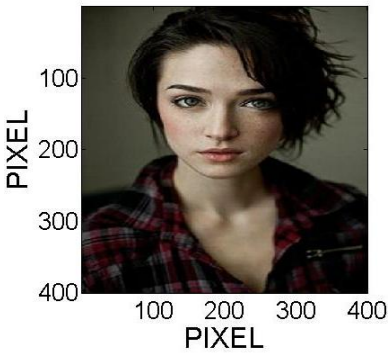
(a) Imagen de entrada



(b) Fase de la llave

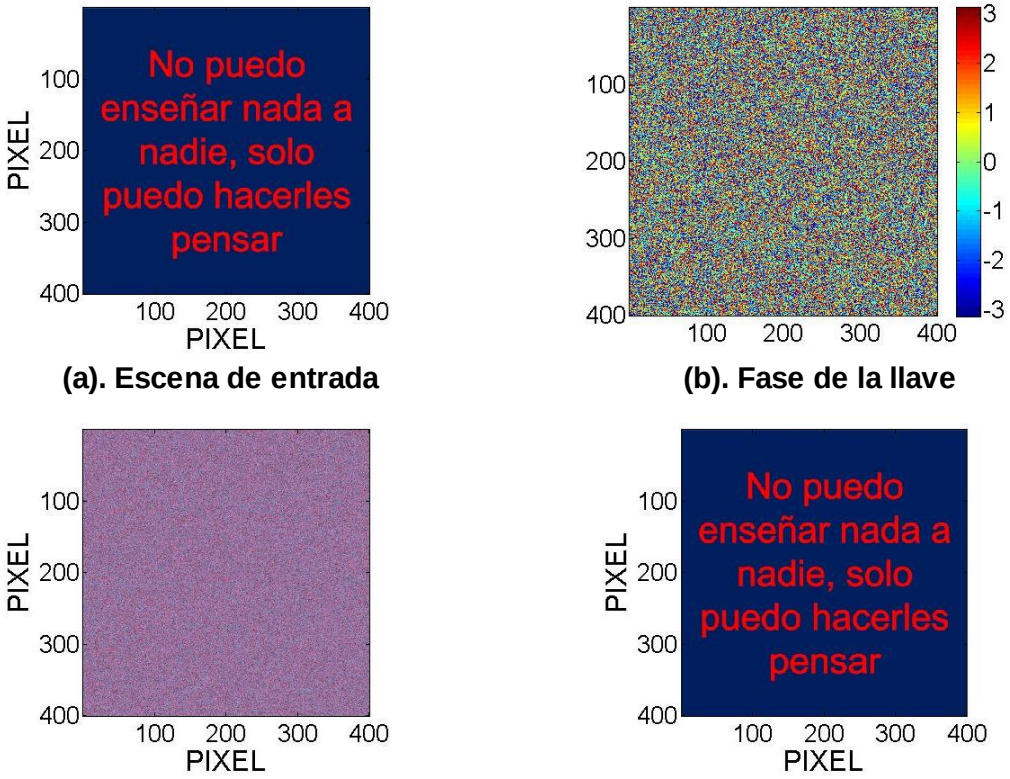


(c) Amplitud de la imagen encriptada



(d) Imagen desencriptada

Figura 2.12. Resultado computacional utilizando un PEVLC y llave en coordenadas cartesianas usando una máscara de fase.



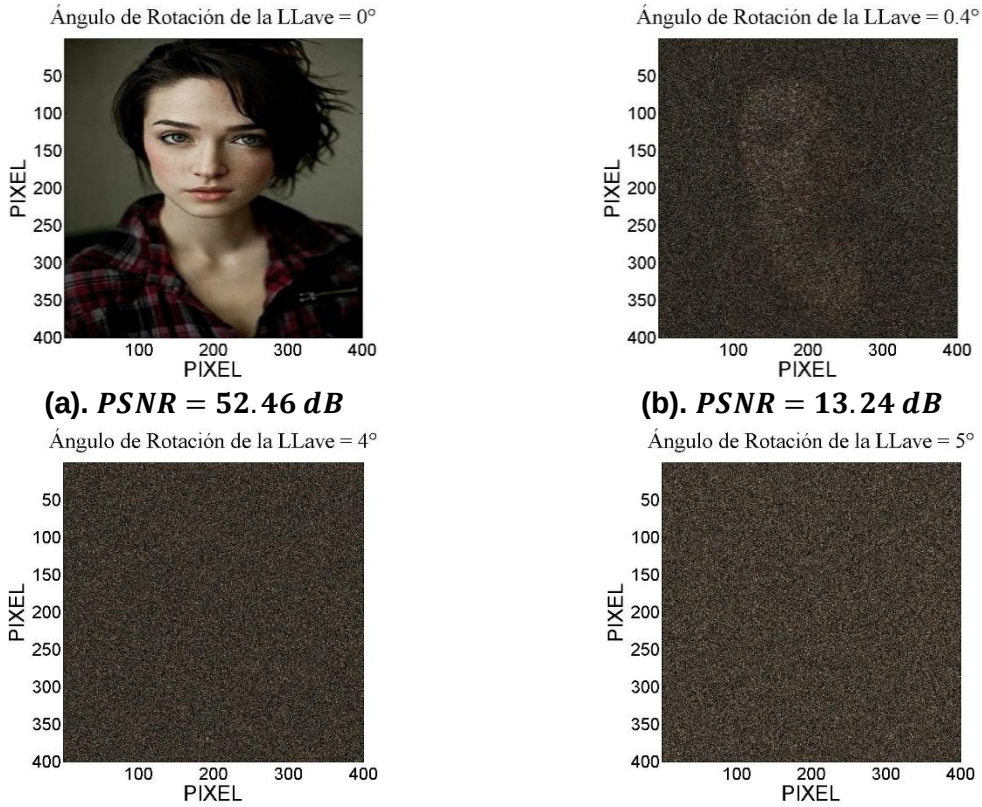
(a). Escena de entrada **(b). Fase de la llave**

(c). Amplitud de la imagen encriptada **(d). Imagen desencriptada**

Figura 2.13. Resultado computacional utilizando un PEVLC y llave en coordenadas cartesianas usando una máscara de fase multiplicando la escena de entrada.

Las **Figura 2.14** y **Figura 2.16** contienen las imágenes desencriptadas con rotación de la llave. En ambos casos se observa que la tolerancia a la rotación de la llave es inferior a ±0.2°. En 0.4° de rotación la distorsión es muy alta en ambos casos, siendo evidente que se debe al uso de la máscara multiplicando la entrada. De otra parte, las **Figura 2.15** y **Figura 2.17** son las correlaciones entre la PSNR y el ángulo de rotación de la llave, de los resultados de las **Figura 2.14** y **Figura 2.16**, respectivamente.

Finalmente, con estos resultados queda demostrada una varianza parcial del proceso de desencriptación con la rotación de la llave en coordenadas cartesianas.



(a). $PSNR = 52.46\ dB$ **(b).** $PSNR = 13.24\ dB$

(c). $PSNR = 12.60\ dB$ **(d).** $PSNR = 12.30\ dB$

Figura 2.14. Imágenes desencriptadas con rotación de la llave, caso objeto rostro y utilizando máscara de fase multiplicando la escena de entrada.

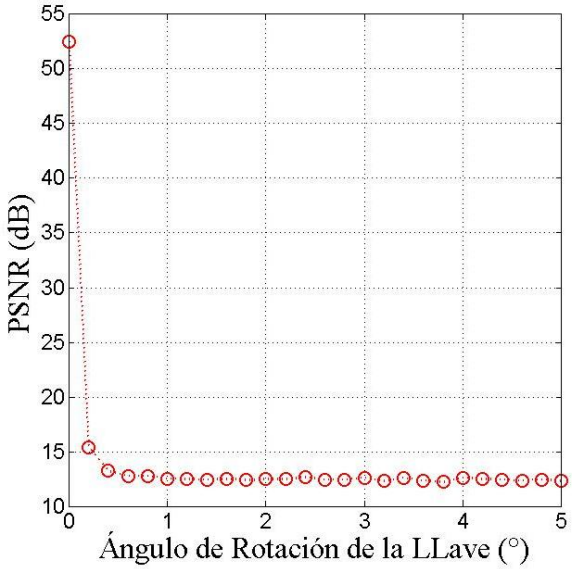
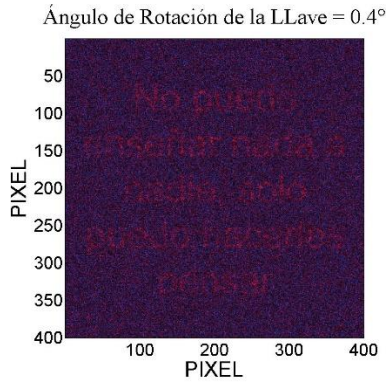
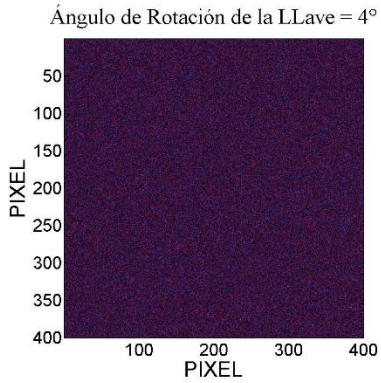


Figura 2.15. Correlación entre la PSNR de imagen desencriptada y el ángulo de rotación de la llave $K_d(u,v)$, caso objeto rostro, utilizando máscara multiplicando la imagen de entrada.

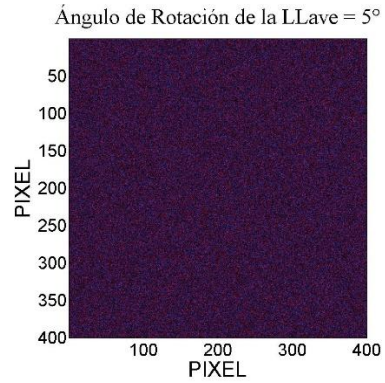
(a). $PSNR = 314.63\ dB$



(b). $PSNR = 13.74\ dB$



(c). $PSNR = 13.12\ dB$



(d). $PSNR = 13.17\ dB$

Figura 2.16. Imágenes desencriptadas con rotación de la llave, caso objeto texto y utilizando máscara de fase multiplicando la escena de entrada.

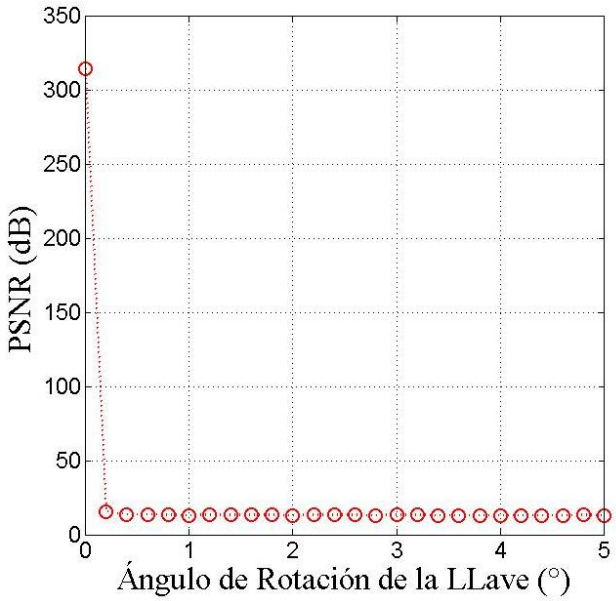
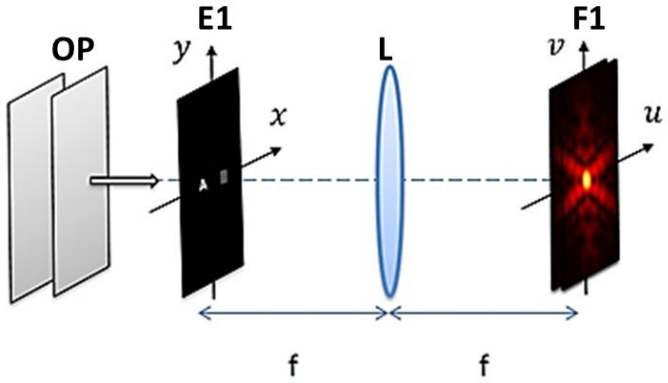


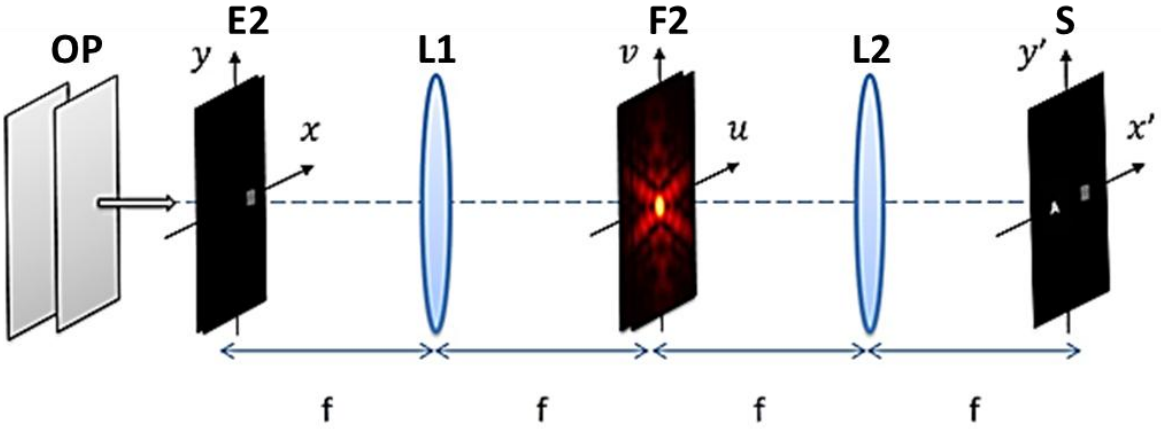
Figura 2.17. Correlación entre la PSNR de imagen desencriptada y el ángulo de rotación de la llave $K_d(u,v)$, caso objeto texto, con máscara multiplicando la escena de entrada.

2.3. ANALISIS FÍSCO-MATEMÁTICO DEL PROCESADOR PEJTC

La arquitectura de transformada conjunta (JTC) al igual que el VLC se utiliza en el reconocimiento de patrones, y también es objeto de estudio en encriptación de imágenes. El procesador de encriptación JTC (PEJTC), a diferencia del PEVLC, requiere un procesador 2f para encriptar (**Figura 2.18(a)**), y un procesador 4f para desencriptar (**Figura 2.18(b)**).



(a). ***Arreglo de encriptación. E1 plano de entrada de la escena a encriptar más llave de encriptación; F1 plano de Fourier (imagen encriptada). L es una lente convergente.***



(b). ***Arreglo de desencriptación. E2 plano llave de desencriptación; F2 plano de Fourier donde se posiciona la imagen encriptada; S plano de la imagen desencriptada. L1 y L2 son lentes convergentes.***

Figura 2.18. Esquema de funcionamiento del PEJTC. OP: onda plana monocromática. f = longitud focal de las lentes L, L1 y L2.

La encriptación se obtiene en el plano F1 debido a la lente L, la cual genera la transformada de Fourier del plano E1. Esta transformada es la superposición de las transformadas de Fourier conjuntas (imagen a encriptar y llave de encriptación). Luego la densidad espectral del plano F1 (se registra, p.e., mediante una cámara digital) se direcciona al plano F2 del arreglo 4f

Figura 2.18(b) (p.e, este plano puede ser un modulador espacial de cristal líquido). Así, con la ubicación de la llave en el plano E2, la lente L1 produce la transformada de Fourier en F2, y entonces se obtiene el producto entre la transformada de la llave y la densidad espectral de la imagen encriptada. Luego, debido a la lente L2 observaremos en el plano S la imagen desencriptada, como consecuencia de la convolución entre la llave y la transformada de Fourier de la imagen encriptada.

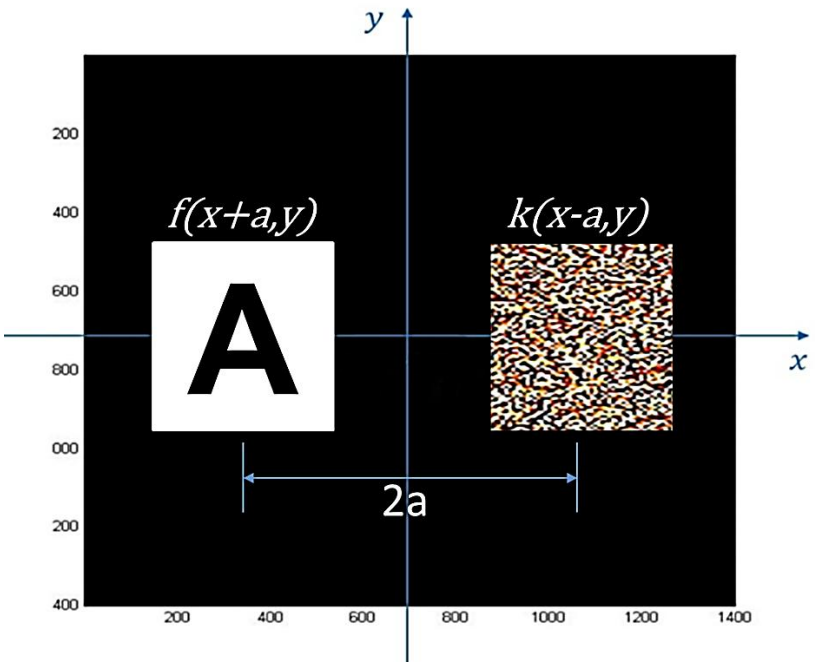


Figura 2.19. Vista del plano de entrada E1 de la Figura 2.18(a).

A continuación hacemos un análisis detallado físico-matemático del PEJTC. Como se ha indicado, el plano de entrada E1 (**Figura 2.19**) contiene la imagen a encriptar $\boldsymbol{f(x+a,y)}$ y la llave de encriptación $\boldsymbol{k(x-a,y)}$, con una separación de 2a entre sí; en términos matemáticos esto es:

$$\boldsymbol{h(x,y) = f(x+a,y) + k(x-a,y).} \qquad \textbf{2.13}$$

Luego el resultado sobre el plano F1 debido a la transformación de Fourier que produce la lente L, se puede representar mediante la **Ec.2.14**.

$$\boldsymbol{H(u,v) = F(u,v)e^{2\pi aui} + K(u,v)e^{-2\pi aui}} \qquad \textbf{2.14}$$

La **Ec.2.15** evidencia que la transformada conjunta está modulada por franjas de interferencia.

$$\boldsymbol{I(u,v) = |H(u,v)|^2}$$

$$\begin{aligned}\boldsymbol{I(u,v)} &= \boldsymbol{\{F(u,v)e^{2\pi aui} + K(u,v)e^{-2\pi aui}\}\cdot\{F^*(u,v)e^{-2\pi aui} + K^*(u,v)e^{2\pi aui}\}} \\ &= \boldsymbol{\{F(u,v)\cdot F^*(u,v) + K(u,v)\cdot F^*(u,v)\cdot e^{-4\pi aui} + F(u,v)\cdot K^*(u,v)\cdot e^{4\pi aui}} \\ &\qquad \boldsymbol{+ K(u,v)\cdot K^*(u,v)\}}\end{aligned} \qquad \textbf{2.15}$$

Una vez direccionada $\boldsymbol{I(u,v)}$ al plano F2 del arreglo 4f (**Figura 2.18(b)**), y posicionada la llave en el plano E2 del mismo arreglo ($\boldsymbol{k(x-a,y)}$), entonces debido a la lente L1 se produce en el plano F2 el producto dado por la **Ec.2.16**.

$$\boldsymbol{I(u,v)\cdot\Im\{k(x-a,y)\} = I(u,v)\cdot K(u,v)e^{-2\pi aui}} \qquad \textbf{2.16}$$

$$\begin{aligned}&\boldsymbol{I(u,v)\cdot K(u,v)e^{-2\pi aui}} \\ &= \boldsymbol{F(u,v)F^*(u,v)K(u,v)e^{-2\pi aui}} \\ &+ \boldsymbol{K(u,v)\cdot F^*(u,v)\cdot K(u,v)e^{-2\pi(3a)ui} + F(u,v)\cdot K^*(u,v)\cdot K(u,v)e^{2\pi aui}} \\ &+ \boldsymbol{K(u,v)\cdot K^*(u,v)\cdot K(u,v)e^{-2\pi aui}}\end{aligned} \qquad \textbf{2.17}$$

Teniendo en cuenta que la llave es de solo fase, el producto $\boldsymbol{K(u,v)K^*(u,v)=1}$ sí se cumple que las llaves encriptación/desencriptación son la misma. Entonces la **Ec.2.17** se puede escribir como:

$$\begin{aligned} &\boldsymbol{I(u,v)\cdot K(u,v)e^{-2\pi aui}} \\ &\boldsymbol{=F(u,v)F^*(u,v)K(u,v)e^{-2\pi aui}} \\ &\boldsymbol{+K(u,v)\cdot F^*(u,v)\cdot K(u,v)e^{-2\pi(3a)ui}+F(u,v)\cdot e^{2\pi aui}+K(u,v)e^{-2\pi aui}} \end{aligned} \qquad \textbf{2.18}$$

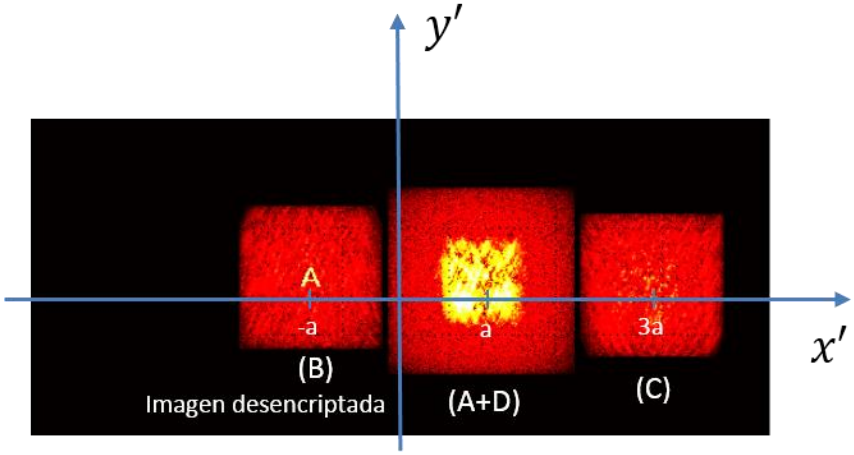
Finalmente, mediante la lente L2 se produce en el plano S la transformada de Fourier de la **Ec.2.18**, esto es $\boldsymbol{f_d(x',y')=\mathfrak{F}^{-1}\{I(u,v)\cdot K(u,v)e^{-2\pi aui}\}}$. El tercer término de la **Ec.2.19** es la imagen desencriptada.

$$\begin{aligned} \boldsymbol{f_d(x',y')=f(x',y')\circledcirc f(x',y')*k(x'-a,y')+} \\ \boldsymbol{+k(x',y')\circledcirc f(x',y')*k(x'-3a,y')+f(x'+a,y')+k(x'-a,y').} \end{aligned} \qquad \textbf{2.19}$$

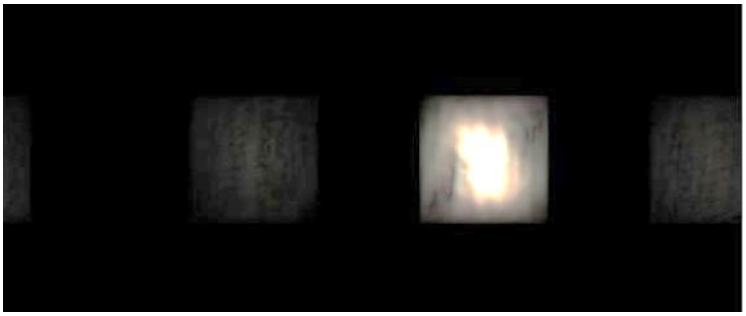
Tabla 3 Descripción física de los términos del plano S del PEJTC.

$\boldsymbol{A=f(x',y')\circledcirc f(x',y')*k(x'-a,y')}$	Autocorrelación de la imagen de entrada, convolucionada con la llave. Este término aparece en x'=a.
$\boldsymbol{B=f(x'+a,y')}$	Imagen desencriptada en la posición x'=-a.
$\boldsymbol{C=k(x',y')\circledcirc f(x',y')*k(x'-3a,y')}$	Correlación cruzada entre la llave y la imagen de entrada, convolucionada con la misma llave. Esta energía aparece en x'=**3a**.
$\boldsymbol{D=k(x'-a,y')}$	Representa la llave ubicada en x'=a, es decir superpuesta con el término A.

En la **Figura 2.20(a)** presentamos la imagen del plano S, resultado de la modelación computacional del PEJTC; este es el resultado de encriptación de la letra **A**. Sobre la imagen del plano S se identifican los términos A, B, C y D descritos en la **Tabla 3**. Los términos A, C y D representan ruido, los cuales no son de interés.



(a). Resultado con imagen binaria letra A.



(b). Resultado con imagen RGB rostro.

Figura 2.20. Resultado computacional de encriptación de la letra A. Intensidad del plano S del PEJTC. DC se define como fondo continuo que aparece con el cálculo de la intensidad del plano S.

Se hicieron pruebas con imágenes complejas como las utilizadas con el PEVLC, sin embargo, los resultados fueron infructuosos. La mayor dificultad encontrada se dio en el proceso de desencriptación debido a que no encontramos el método para reducir la fuerte distorsión en la imagen de salida; en la **Figura 2.20(b)** se muestra

un resultado, razón por la cual y por limitaciones de tiempo, decidimos no estudiar la varianza a la rotación con el PEJTC, y se deja como perspectiva para futuros trabajos.

Capítulo III

3. LLAVES DE SOLO FASE EN CHC Y ESTUDIO DE LA VARIANZA EN LA DESENCRIPTACIÓN CON LA ROTACIÓN DE LA LLAVE

En este capítulo presentamos una solución a la varianza que se presenta en la desencriptación debida a la rotación de la llave durante este proceso, varianza que ha sido discutida en el capítulo anterior. La solución que presentamos se refiere al uso de llaves de solo fase en CHC (*Circular harmonic components*); demostramos que es una solución donde se disminuye significativamente el nivel de varianza de la desencriptación. En parte la idea ha surgido de los correladores ópticos invariantes a la rotación de los objetivos en la escena, donde se usan filtros armónicos circulares [47, 48]. Demostramos entonces que descomponiendo la fase de la llave en componentes armónicos circulares CHC se aumenta significativamente la tolerancia a dicha rotación de la llave, durante la desencriptación de la imagen. El estudio se hizo utilizando un arreglo PEVLC. En el siguiente apartado discutimos en que consiste la descomposición en CHC, y mostramos como es el proceso computacional para obtener este tipo de descomposición.

3.1. DESCOMPOSICIÓN EN COMPONENTES ARMÓNICAS CIRCULARES

Sea la imagen $\boldsymbol{f}(\boldsymbol{x},\boldsymbol{y})$ en coordenadas cartesianas. Entonces esta imagen se puede descomponer en CHC mediante una serie de Fourier según la siguiente relación:

$$\boldsymbol{f_{CHC}}(\boldsymbol{r},\boldsymbol{\theta}) = \sum_{\boldsymbol{m}=-\infty}^{\infty} \boldsymbol{f_m}(\boldsymbol{r})\boldsymbol{e}^{\boldsymbol{im\theta}} \qquad \textbf{3.1}$$

Con,

$$f_m(r) = \frac{1}{2\pi}\int_0^{2\pi} f(r,\theta)e^{-im\theta}\, d\theta \qquad \textbf{3.2}$$

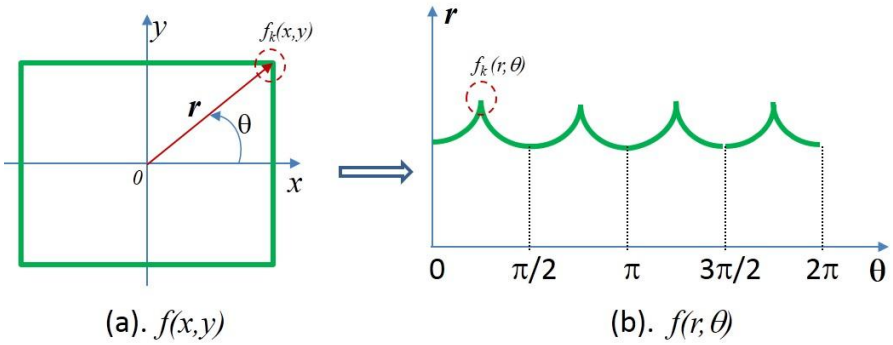
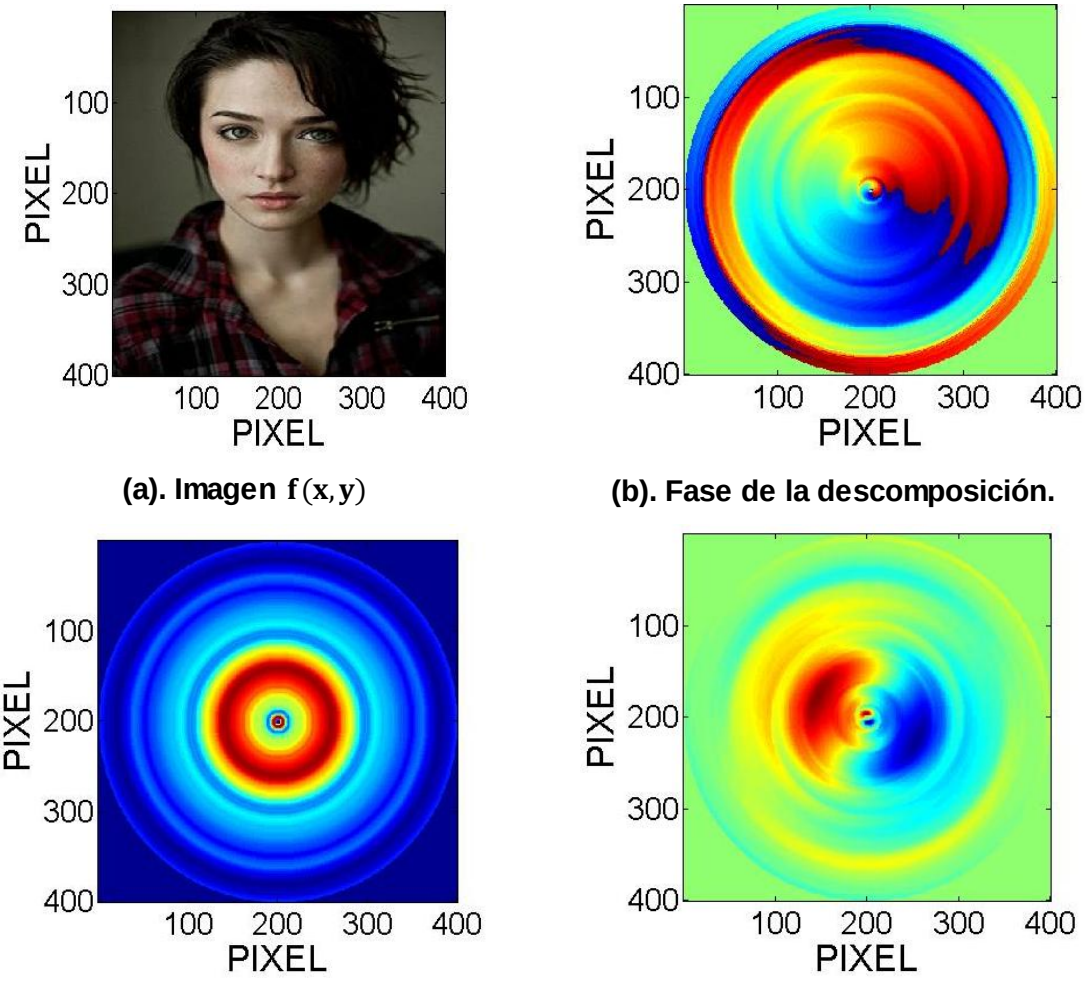


Figura 3.1. Transformación de $f(x,y)$ a coordenadas polares $f(r,\theta)$.

La **Ec.3.2** permite calcular los pesos $f_m(r)$ de orden m. Donde r es la componente radial y θ la componente angular de la función en coordenadas polares. m es el orden entero de la descomposición en armónicos circulares. Para aplicar la **Ec.3.2** es necesario transformar la imagen $f(x,y)$ a coordenadas polares $f(r,\theta)$ [49], tal como sugiere el esquema de la **Figura 3.1**, para el caso de aro rectangular. Cada valor $f_k(x,y) = f_k(r,\theta)$ es reubicado en el plano polar en su respectiva coordenada (r,θ). Nosotros adaptados el algoritmo, para generar armónicos circulares, propuesto por los autores de la Ref. [50], que ellos utilizaron en el diseño de filtros para reconocimiento de patrones. Utilizamos el algoritmo CHC para los resultados de la **Figura 3.2** que es la descomposición CHC de orden $m = 1$, de la imagen **Figura 3.2(a)**; Las **Figura 3.2(b)-(d)** son la fase, parte real y módulo de la descomposición, respectivamente.



(a). Imagen $f(x,y)$ **(b). Fase de la descomposición.**

(c). Parte real de la descomposición. **(d). Módulo de la descomposición.**

Figura 3.2 Descomposición en armónico circulares (orden m=1).

3.1.1. Llave de solo fase en CHC, a partir de una distribución de fase en coordenadas cartesianas.

Sea la distribución de fase aleatoria definida en el espacio de frecuencias como $\boldsymbol{\Phi(u,v)}$. Entonces, según las **Ec.3.1** y **3.2**, la fase en CHC se calcula mediante la serie de Fourier,

$$\boldsymbol{\Phi_{CHC}(\rho,\phi) = \sum_{m=-\infty}^{\infty} \Phi_{(m+f)}(\rho) e^{-i(m+f)\phi}}, \qquad \textbf{3.3}$$

Donde,

$$\boldsymbol{\Phi_{(m+f)}(\rho) = \frac{1}{2\pi}\int_0^{2\pi} \Phi(\rho,\phi) e^{-i(m+f)\phi} d\phi} \qquad \textbf{3.4}$$

Siendo $(\boldsymbol{\rho,\phi})$ las componentes de frecuencias en coordenadas polares, $\boldsymbol{\Phi(\rho,\phi)}$ la fase en coordenadas polares y $\boldsymbol{\Phi_{CHC}(\rho,\phi)}$ la fase en armónicos circulares. Aquí introducimos el concepto de orden fraccional $\boldsymbol{f = [0,1]}$ de la descomposición ahora podemos reescribir el resultado como:

$$\boldsymbol{\Phi_{CHC}(\rho,\phi)_{m+f} = |\Phi_{CHC}(\rho,\phi)| e^{i\Psi(\rho,\phi)}} \qquad \textbf{3.5}$$

Donde $\boldsymbol{\Psi(\rho,\phi)}$ es la fase en armónicos circulares. Entonces la nueva llave de encriptación de solo fase en CHC la obtenemos dividiendo la **Ec.3.5** entre su módulo, esto es:

$$\boldsymbol{K_{CHC_{m+f}} = e^{i\Psi(\rho,\phi)}} \qquad \textbf{3.6}$$

3.1.2 Explicación del algoritmo de descomposición en armónicos circulares

El algoritmo fue adaptado en lenguaje MATLAB. Para explicar su funcionamiento de cálculo de las componentes armónicas, utilizamos una imagen de 50x50 pixeles que contiene un aro cuadrado (**Figura 3.3**), y que representamos por $f(x,y)$. La primera parte del algoritmo transforma la imagen de coordenadas cartesianas a coordenadas polares $f(x,y) \rightarrow f(r,\theta)$.

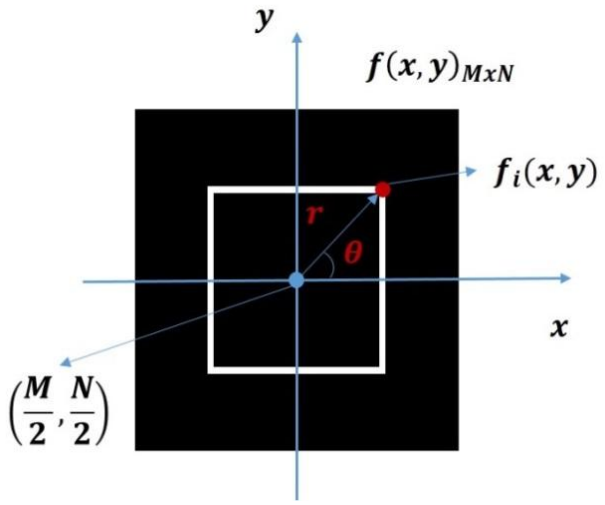
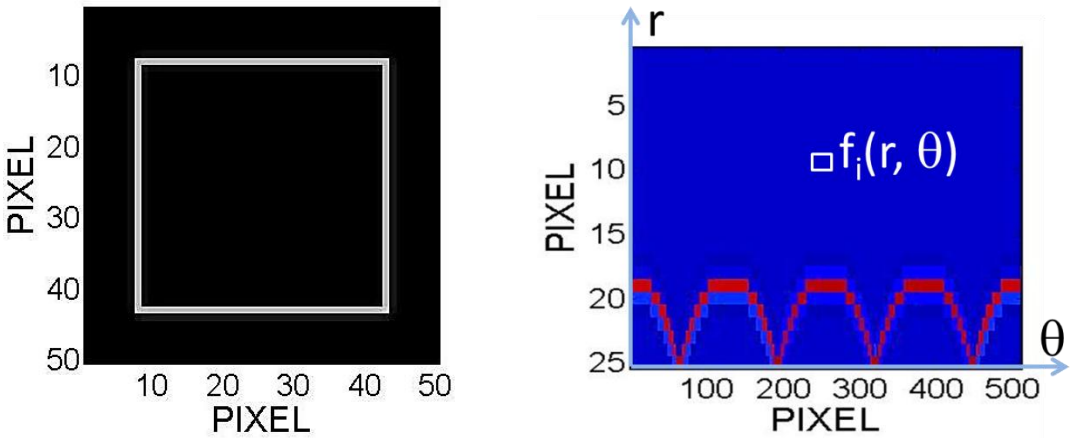


Figura 3.3. Imagen digital $f(x,y)$ en coordenadas cartesianas.



(a). Imagen en coordenadas cartesianas.

(b). Imagen en coordenadas polares $f(r,\theta)$.

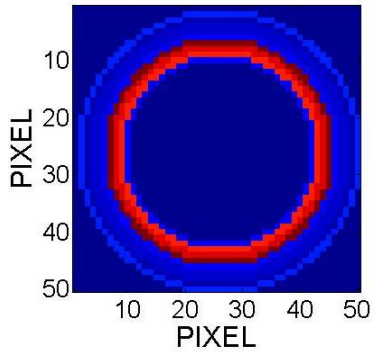
Figura 3.4. Resultado computacional de transformación de la imagen (a) a polares.

Teniendo en cuenta que la imagen es una matriz, cuyo origen está en la esquina superior izquierda, este se debe trasladar al centro de la matriz $\left(\frac{M}{2}, \frac{N}{2}\right)$. Entonces se inicia, de manera radial el cálculo de la posición polar de cada nivel de intensidad haciendo un barrido angular de $\mathbf{0}$ a $\mathbf{2\pi}$ radianes, y se obtiene la matriz $\boldsymbol{f(r,\theta)}$.

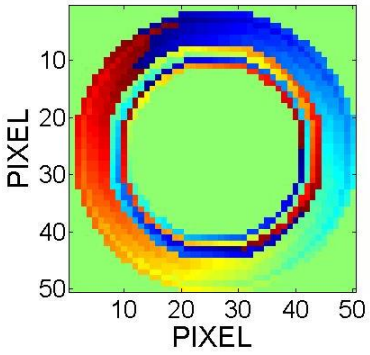
El resultado computacional final de transformación a polares de la imagen $\boldsymbol{f(x,y)}$ se muestra en la **Figura 3.4(b)**. Las posiciones de las columnas en pixeles $[\mathbf{1}:\mathbf{511}]$ representan valores angulares de $\boldsymbol{\theta}$ entre $\left[\mathbf{0}:\frac{\mathbf{2\pi}}{\mathbf{510}}:\mathbf{2\pi}\right]$ radianes. Las posiciones de las filas en pixeles $[\mathbf{1},\mathbf{25}]$ representa la posición radial de cada valor de intensidad $\boldsymbol{f(r,\theta)}$. Calculada la imagen en coordenadas polares, se calculan los coeficientes $\boldsymbol{f(r)_m}$ mediante la expresión discreta de la **Ec.3.2** o **Ec.3.4**,

$$\boldsymbol{f_m(r) = \frac{1}{n}\sum_{k=1}^{n} f\left(\frac{2\pi k}{n}\right) e^{-im\frac{2\pi k}{n}}} \qquad \textbf{3.7}$$

Donde n es número de muestras entre $[\mathbf{0},\mathbf{2\pi}]$. Calculados estos coeficientes se obtiene la matriz final $\boldsymbol{f_{CHC}(r,\theta)}$ aplicando la **Ec.3.1**. La **Figura 3.5** muestra el resultado computacional final de la descomposición CHC de la imagen de la **Figura 3.4(a)**. El ordenamiento de los valores $\boldsymbol{f_{CHC}(r,\theta)}$ en la matriz final se hace con el origen en $\left(\frac{M}{2}, \frac{N}{2}\right)$.

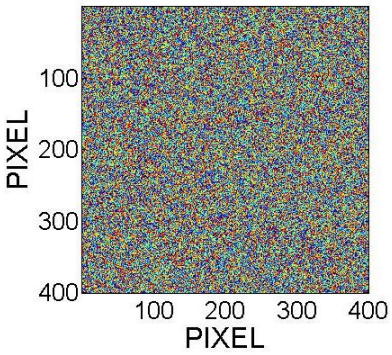


(a).Amplitud de la descomposición en CHC.

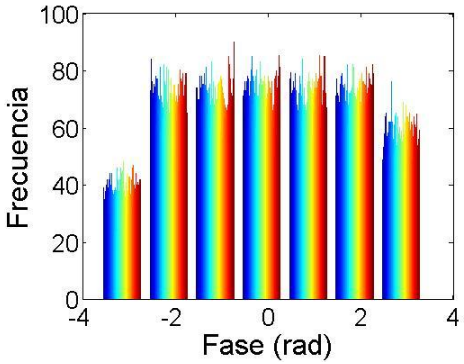


(b). Fase de la descomposición en CHC.

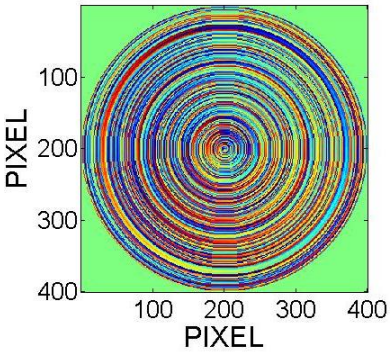
Figura 3.5. Resultado computacional de la descomposición en armónicos circulares.



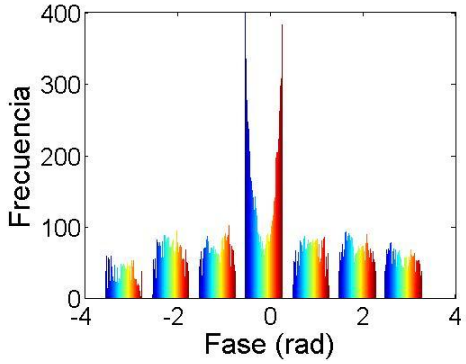
(a). Fase de una llave $\mathbf{K(u,v)}$



(b). Histograma de (a)



(c). Fase de una llave $\mathbf{K_m(\rho,\phi)}$ con $m = 1$.

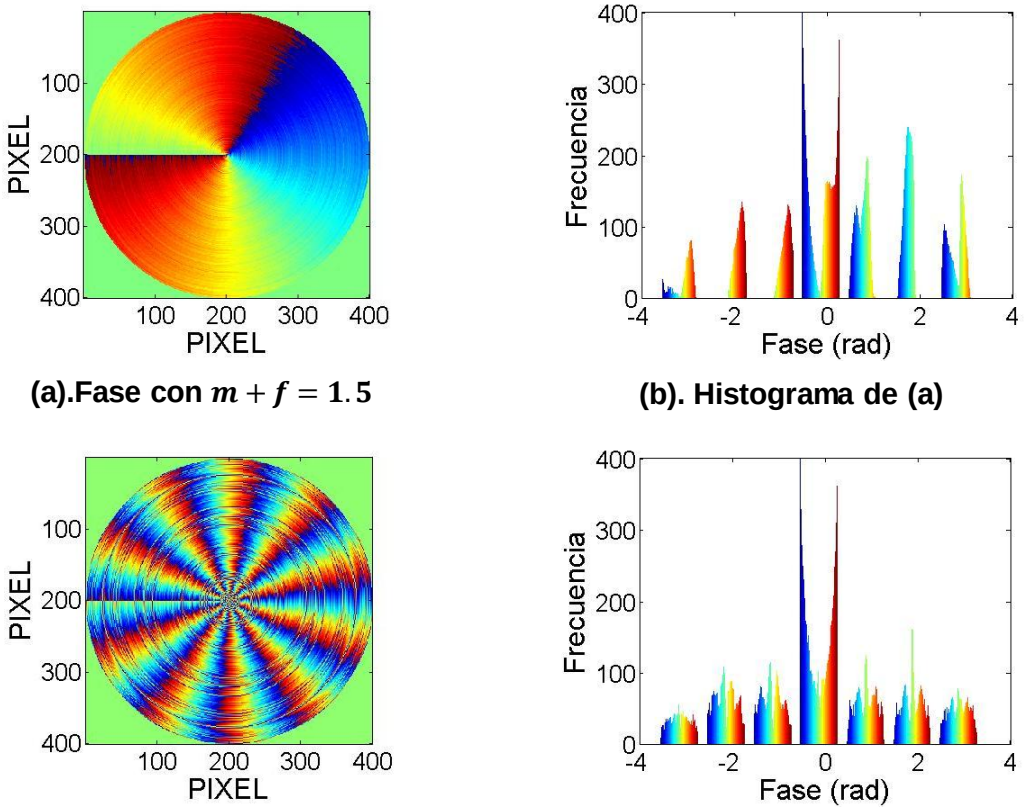


(d). Histograma de (c)

Figura 3.6. Fase e histogramas de llaves tipo $K(u,v)$ y $K_m(\rho,\phi)$.

Los resultados de la **Figura 3.6** son distribuciones de fase aleatorias en coordenadas cartesianas, y su respectiva fase calculada en armónicos circulares. Ambos histogramas presenta una distribución uniforme de valores de fase entre $[-\pi, \pi]$, indicador de que las dos distribuciones de fase permiten encriptar con un buen nivel de seguridad.

También consideramos la descomposición fraccional en armónicos circulares de la misma fase de la **Figura 3.6(a)** (ver resultado en la **Figura 3.7**).



(a).Fase con $m + f = 1.5$ **(b). Histograma de (a)**

(c). Fase con $m + f = 10.5$ **(d). Histograma de (c).**

Figura 3.7. Fase con descomposición fraccional armónica circular.

El histograma de la llave para un orden de $m + f = 1.5$ (**Figura 3.7(b)**) es un ejemplo de distribución de fase no adecuada para encriptar, por que la distribución

no es suficientemente uniforme y por ende un nivel de encriptación bueno, debe tener un PSNR menor a 12 dB; para este tipo de llave se encontró niveles de encriptación con PSNR superiores a 20 dB. En cambio una llave con un orden fraccional superior, por ejemplo $\boldsymbol{m = 10.5}$, tiene una mejor distribución de valores de fase entre $[-\boldsymbol{\pi}, \boldsymbol{\pi}]$ (véase **Figura 3.7(d)**).

En el siguiente Capítulo presentamos resultados computacionales de modelación de PEVLC usando este tipo de llaves con fase en componentes armónicas circulares, como una solución a la varianza de la desencriptación con la rotación de la llave.

Capítulo IV

4. MODELACIÓN COMPUTACIONAL DEL PEVLC USANDO LLAVES CHC

En el Capítulo 2 se hizo el análisis físico-matemático del PEVLC, por lo tanto en este apartado no incluimos este aspecto matemático teniendo en cuenta que el uso de llaves en armónicos circulares no cambia el modelo físico-matemático del PEVLC. La diferencia solo está en el algoritmo computacional donde adicionamos el algoritmo que calcula la llave de solo fase en armónicos circulares (algoritmo que ha sido explicado en el literal Cap. III.3.1). La **Figura 4.1** muestra el diagrama de flujo del PEVLC usando llaves en CHC.

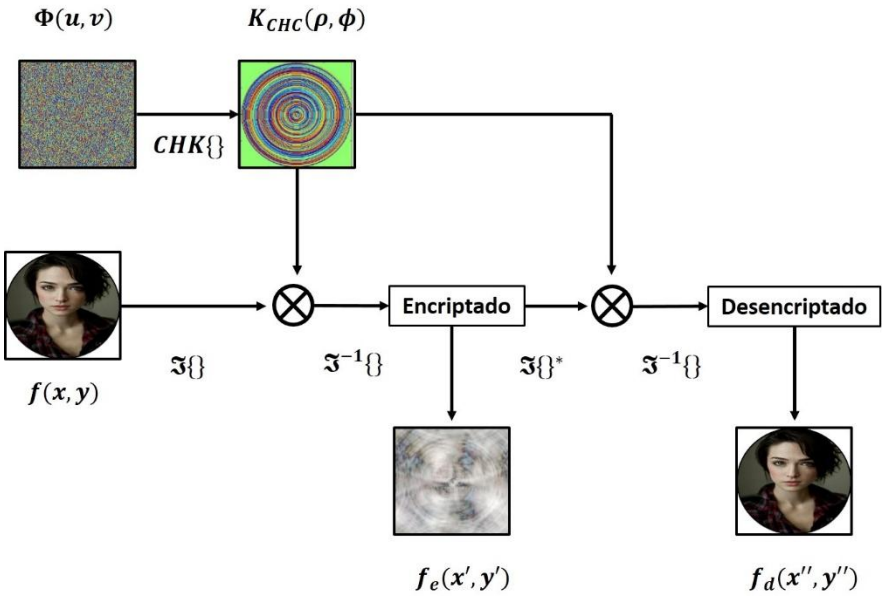


Figura 4.1. Diagrama de flujo del proceso de encriptación/desencriptación de un PEVLC usando una llave en CHC. ⊗ operador multiplicación aritmética; $\mathfrak{I}\{\}$ operador transformada de Fourier; $CHC\{\}$ generación de la llave en CHC.

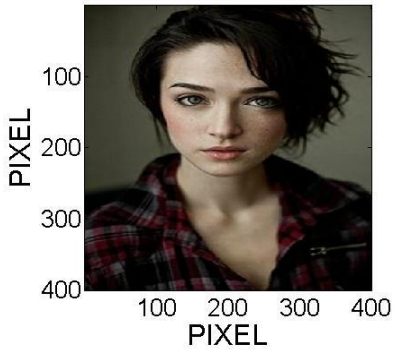
El diagrama de flujo describe las dos etapas del procesador de encriptación PEVLC. La primera etapa es la encriptación, donde a partir de la fase $\boldsymbol{\Phi(u,v)}$ se le aplica el algoritmo CHC para obtener la llave de solo fase $\boldsymbol{K_{CHC}(\rho,\phi)}$. Esta llave se multiplica por la transformada de Fourier de la escena de entrada y a este producto se aplica una transformada de Fourier para así obtener la imagen encriptada. Mediante el proceso inverso se obtiene la imagen desencriptada (la **Figura 4.2** es un ejemplo de los resultados obtenidos computacionalmente). Para ello, la imagen encriptada se conjuga y se calcula la transformada de Fourier, resultado que se multiplica por la llave, para entonces aplicar otra transformada de Fourier cuyo resultado es la imagen desencriptada.

4.1 ESTUDIO DE LA VARIANZA DE LA DESENCRIPTACIÓN CON LA ROTACIÓN DE LA LLAVE

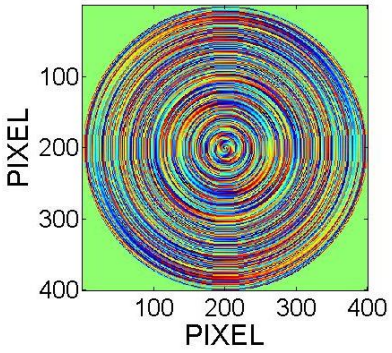
Respecto a los resultados de este estudio de la varianza de la desencriptación con la rotación de la llave CHC, solo presentamos los obtenidos con la escena rostro (formato RGB de 400x400 pixeles). Igualmente hicimos pruebas con otros tipos de escenas de entrada sin que hayamos encontrado diferencias significativas que amerite hacer una discusión en función del tipo de escena de entrada. Dividimos el estudio computacional en tres casos: 1). Sin máscara multiplicando la imagen de entrada, 2). Con pupila circular binaria multiplicando la imagen de entrada y 3). Con máscara de estructura periodica $\boldsymbol{h^{-1}(ax^p,by^p)}$ multiplicando la imagen de entrada. En cada caso la imagen se desencriptó rotando la llave de 0° a 360°, con un paso de 5°; en este informe mostramos cuatro imágenes intermedias de tales resultados, y la correlación PSNR vs Ángulo de rotación de la llave.

CASO I. SIN MÁSCARA MULTIPLICANDO LA ENTRADA.

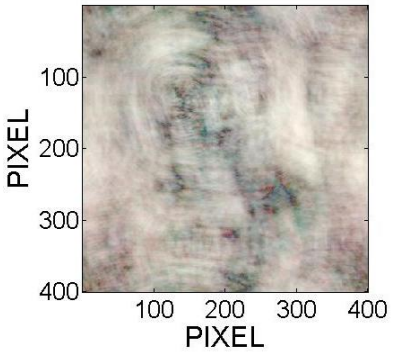
La imagen de la **Figura 4.2(a)** es la escena de entrada, la distribución de fase de la llave es la imagen de la **Figura 4.2(b)**, y la imagen de la **Figura 4.2(c)** es la amplitud de la imagen encriptada. La forma de la imagen encriptada podría resultar sospechosa, y por otro lado consideramos que el nivel de encriptación no es bueno.



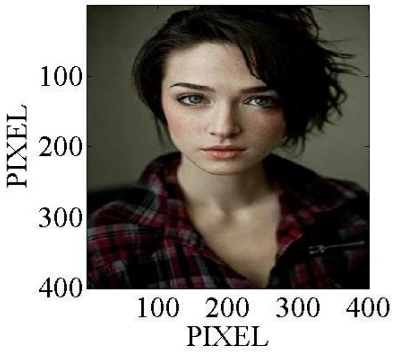
(a). Imagen de entrada



(b). Fase de la llave CHK



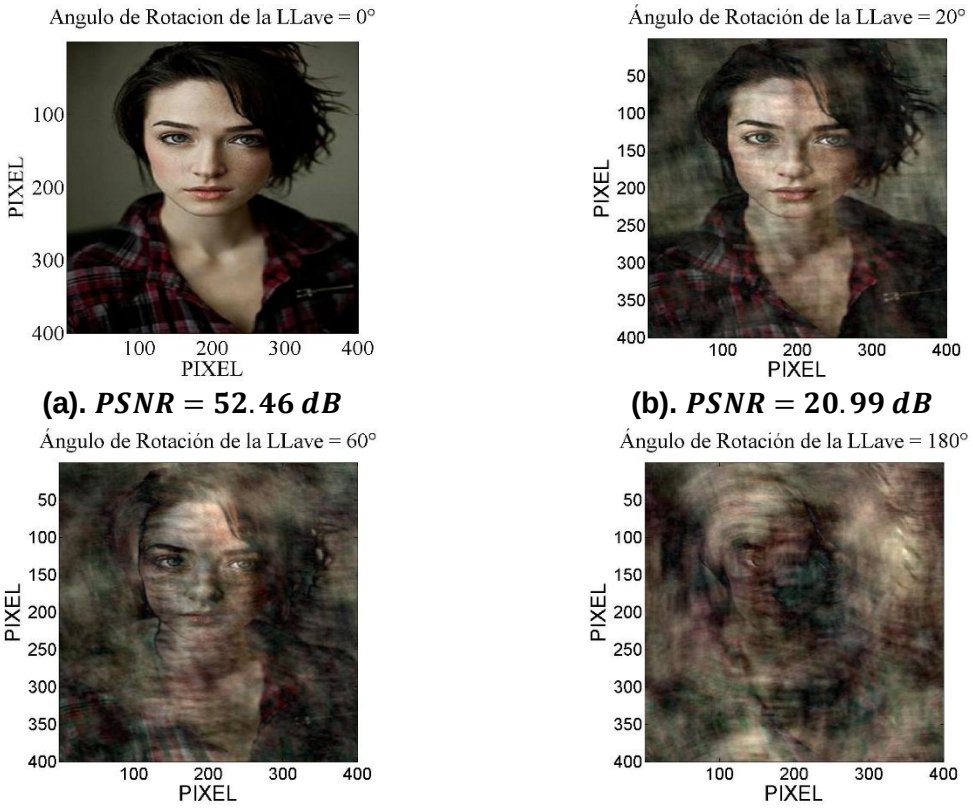
(c). Amplitud de la imagen encriptada



(d). Imagen desencriptada

Figura 4.2. Encriptación/Desencriptación sin máscara multiplicando la entrada. Caso I.

Los resultados de este caso (**Figura 4.3**) muestran una tolerancia a la rotación de la llave de $\pm 60°$, es decir que la imagen ha sido parcialmente desencriptada, en otras palabras el $PSNR = 15.19dB$ se puede considerar aceptable. Un PSNR inferior a 15dB significa alta distorsión de la imagen desencriptada.



(a). $\boldsymbol{PSNR = 52.46\,dB}$ **(b).** $\boldsymbol{PSNR = 20.99\,dB}$

(c). $\boldsymbol{PSNR = 15.19\,dB}$ **(d).** $\boldsymbol{PSNR = 10.89\,dB}$

Figura 4.3. Resultados varianza de la imagen desencriptada, con llave $K_d(\rho,\phi)$, caso I.

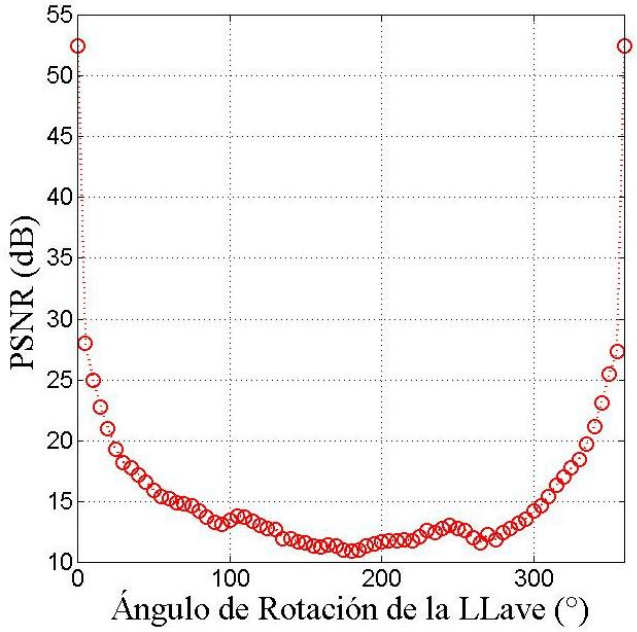
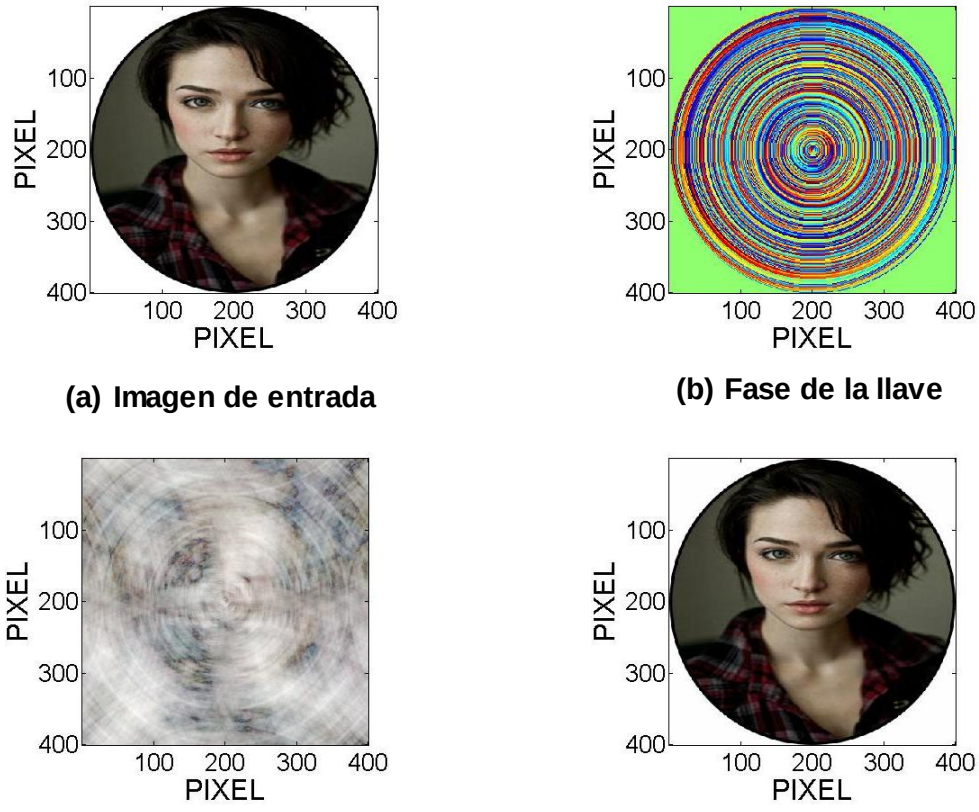


Figura 4.4. PSNR Vs ángulo de rotación de la llave de desencriptación $K_d(\rho,\phi)$. Caso I.

De este caso concluimos que la tolerancia a la rotación de la llave es mayor, en comparación con el análisis hecho en el capítulo II, donde la tolerancia a la rotación de la llave de desencriptación en coordenadas cartesianas $\boldsymbol{K_d(u,v)}$ es menor a ±1°, mientras que para este caso, permite una tolerancia de aproximadamente ±60°. Solo cuando no hay rotación de la llave se obtiene la imagen desencriptada con ruido no perceptible por el ojo humano ($PSNR = 52.46\ dB$).

CASO II. MULTIPLICANDO LA ENTRADA POR UNA PUPILA CIRCULAR DE AMPLITUD BINARIA.



(a) Imagen de entrada

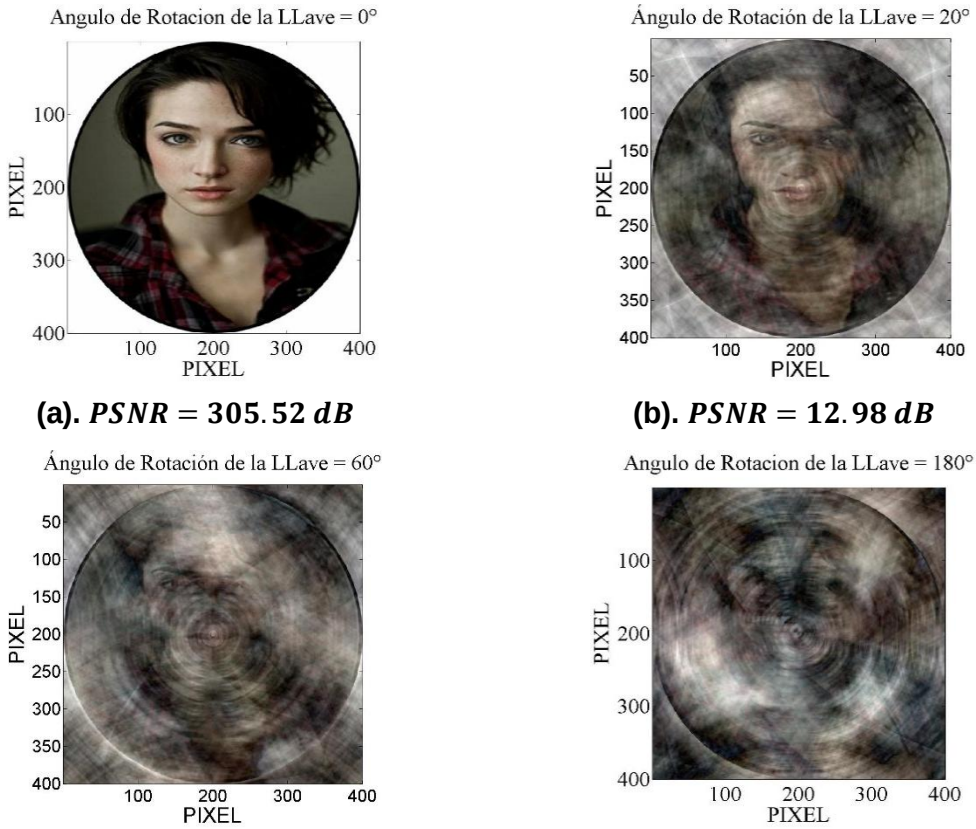
(b) Fase de la llave

(c) Amplitud de imagen encriptada

(d) Imagen desencriptada

Figura 4.5. Encriptación/Desencriptación multiplicando la entrada con una pupila circular binaria. Caso II.

La **Figura 4.5** es el resultado sin rotar la llave de desencriptación. En las **Figura 4.6** y **Figura 4.7** se puede observar que la tolerancia a disminuido con respecto al Caso I, de ±60° a ±20° de ángulo de giro de la llave, que corresponde a una imagen desencriptada con un $PSNR = 12.98dB$ debida a la distorsión causada por el ruido. Consideramos que la inclusión de la máscara (pupila circular binaria), multiplicando el plano de entrada, es la responsable de la reducción de tolerancia a la rotación de la llave, respecto al caso I.



(a). $\boldsymbol{PSNR = 305.52\ dB}$ **(b).** $\boldsymbol{PSNR = 12.98\ dB}$

(c). $\boldsymbol{PSNR = 8.78\ dB}$ **(d).** $\boldsymbol{PSNR = 8.46\ dB}$

Figura 4.6. Resultados varianaza de la imagen desencriptada, con llave $K_d(\rho,\phi)$, caso II.

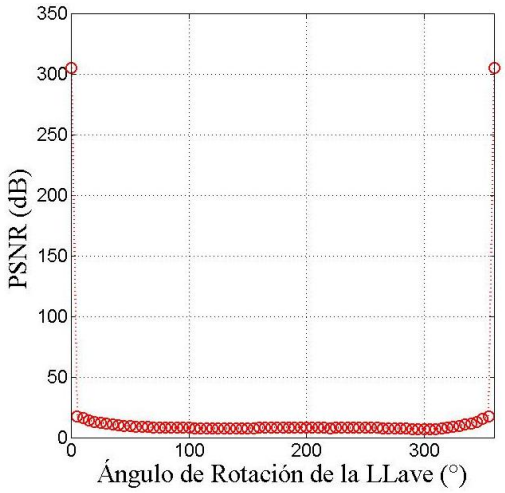
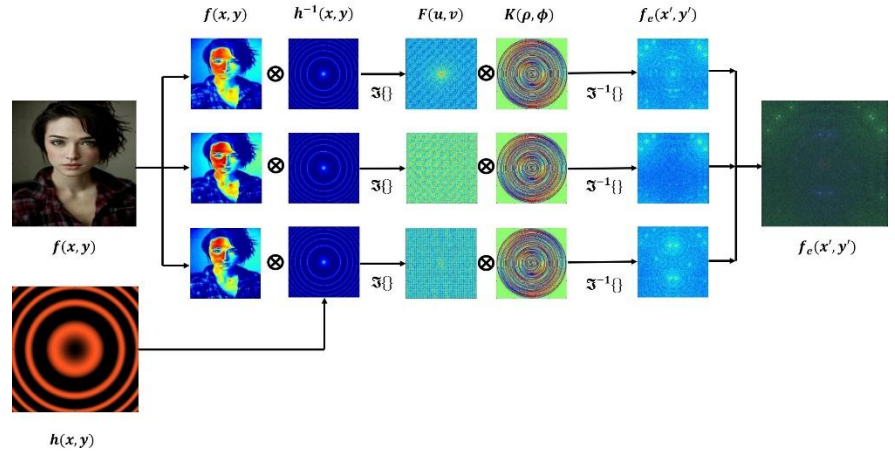


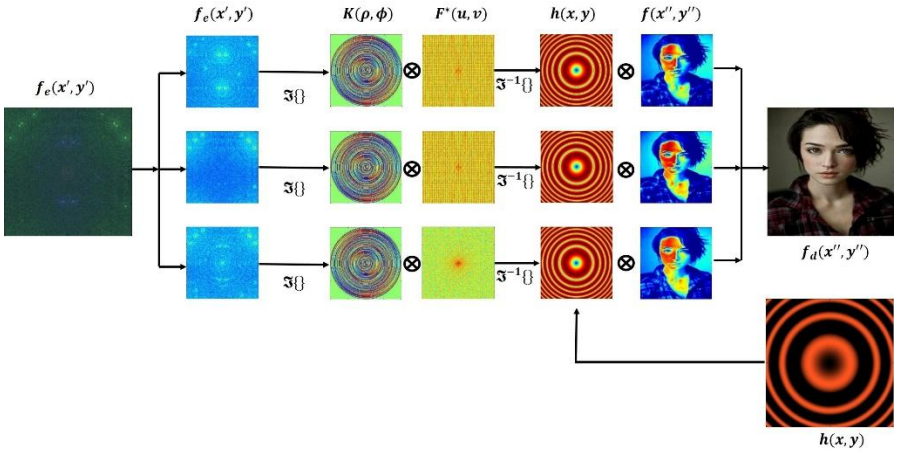
Figura 4.7. PSNR Vs ángulo de rotación de la llave de desencriptación $K_d(\rho,\phi)$. Caso II.

CASO III. MULTIPLICANDO LA ENTRADA POR UNA MÁSCARA DETERMINÍSTICA DE AMPLITUD $h^{-1}(ax^p,by^p)$.

En este caso multiplicamos la imagen de entrada por una máscara determinística de estructura periódica $\boldsymbol{h^{-1}(ax^p,by^p)}$, donde a, b y p son constantes. El diagrama de flujo del algoritmo computacional, de encriptación y desencriptación es el de la **Figura 4.8**; En la etapa de encriptación, primero se genera la máscara $\boldsymbol{h^{-1}(ax^p,by^p)}$ y se multiplica por cada canal RGB, y se aplica una transformada de Fourier para cada canal; este resultado se multiplica por la llave de encriptación $\boldsymbol{K_e(\rho,\phi)}$, y se ejecuta una transformada de Fourier al resultado de cada canal; la salida es la concatenación de los canales RGB, es decir la imagen encriptada $\boldsymbol{f_e(x',y')}$ (**Figura 4.8(a)**).



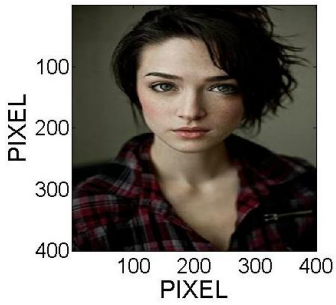
(a). Etapa de encriptación.



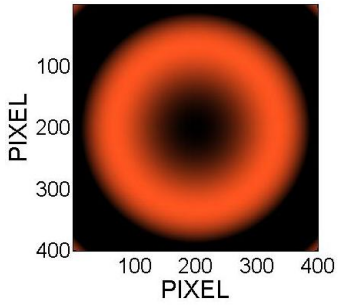
(b). Etapa de desencriptación.

Figura 4.8. Encriptación caso III. $F(u,v)$ transformada del producto entre la imagen y la máscara; $\Im\{\}$ operador transformada de Fourier; $\otimes$ operador multiplicación aritmética.

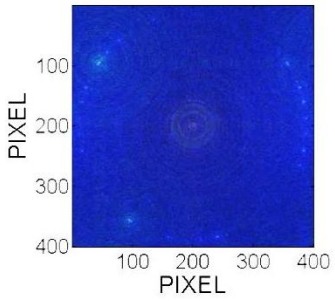
La etapa de desencriptación es el proceso inverso, excepto que antes de concatenar los canales RGB se multiplican por $\boldsymbol{h(ax^p, by^p)}$. La **Figura 4.9** muestra los resultados de encriptación/desencriptación de la imagen de entrada rostro, usando la máscara de la **Figura 4.9(b)**, y una llave con CHC de orden $m = 1$. Se puede observar en la **Figura 4.9(c)** que la imagen encriptada presenta un patrón determinístico, diferente a los obtenidos en los casos anteriores; sin embargo, encontramos que el nivel tolerancia a la rotación de la llave es nulo, es decir que el proceso desencriptación es totalmente variante a la rotación de la llave; resultados que podemos analizar en la **Figura 4.10**.



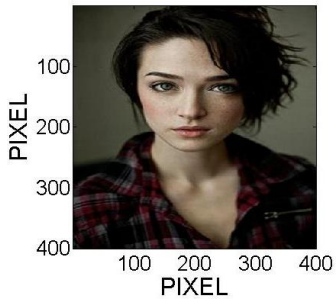
(a). Imagen de entrada.



(b). Máscara para $a = b = 1$; $p = 2$



(c). Amplitud de la imagen encriptada



(d). Imagen desencriptada

Figura 4.9. Encriptación/Desencriptación multiplicando la entrada con máscaras de simetría circular. Caso III.

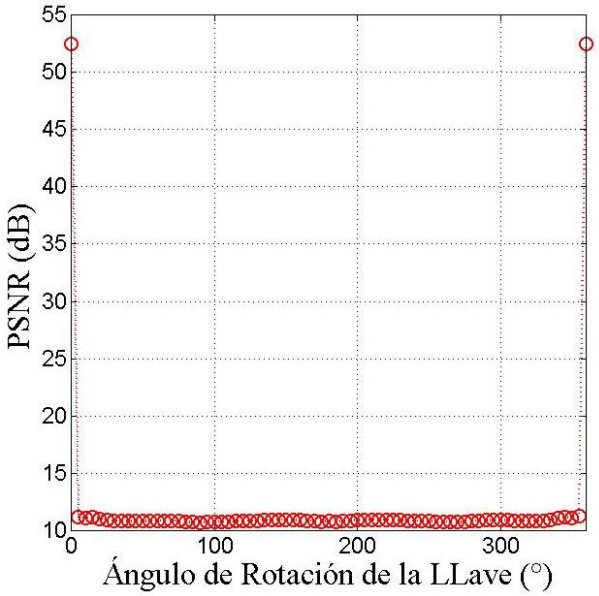
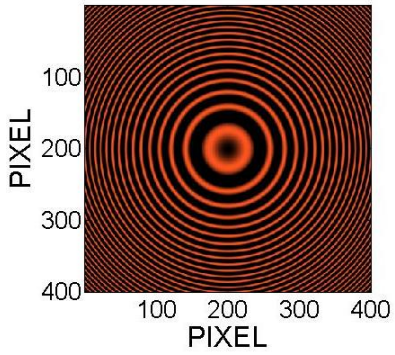
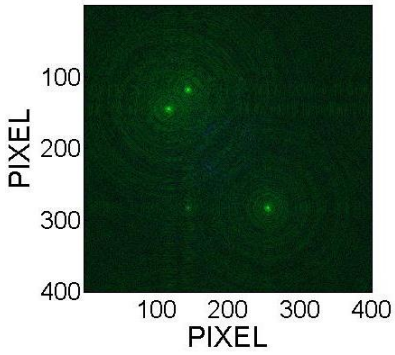


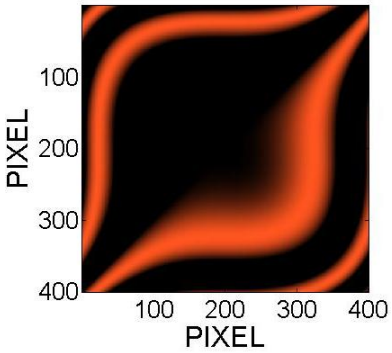
Figura 4.10. PSNR Vs ángulo de rotación de la llave de desencriptación $K_d(\rho, \phi)$, m=1. Caso III.



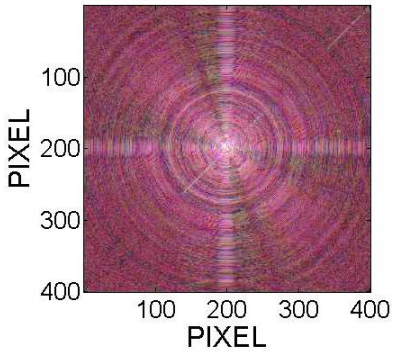
(a). Máscara $a = b = 0.5$; $p = 2$



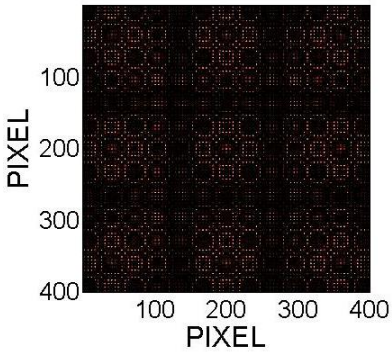
(b). Amplitud de $f_e(x', y')$



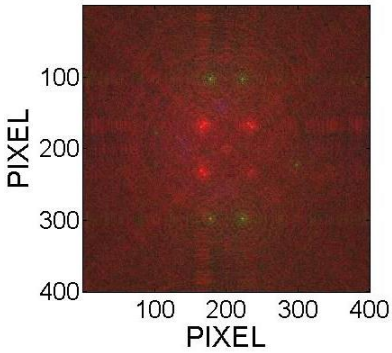
(c) Máscara $a = b = 0.1$; $p = 3$



(d). Amplitud de $f_e(x', y')$

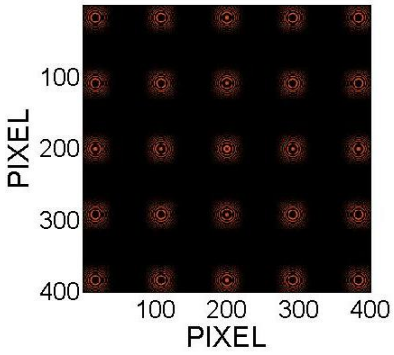


(e). Máscara $a = b = 15$; $p = 2$

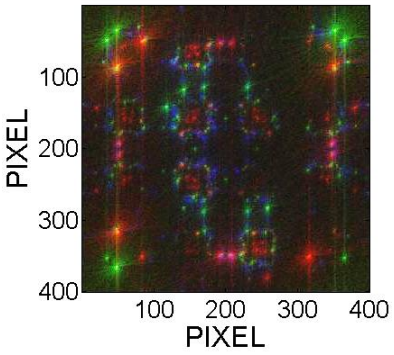


(f). Amplitud de $f_e(x', y')$

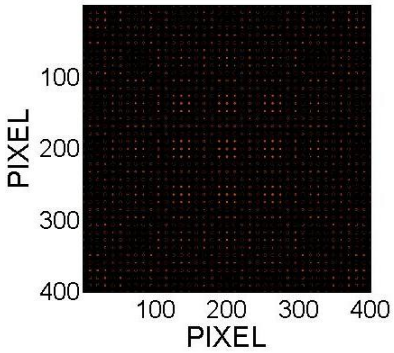
Figura 4.11. Camuflaje del criptograma multiplicando la entrada con máscaras de simetría circular. Caso III. Llave con CHC $m = 1$.



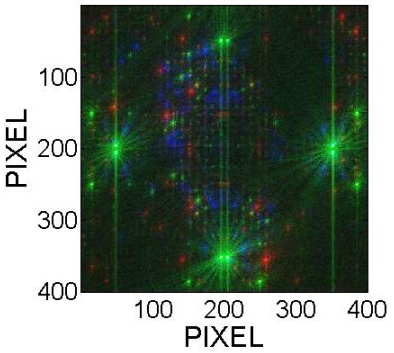
(a). Máscara $a = b = 3.4$; $p = 2$



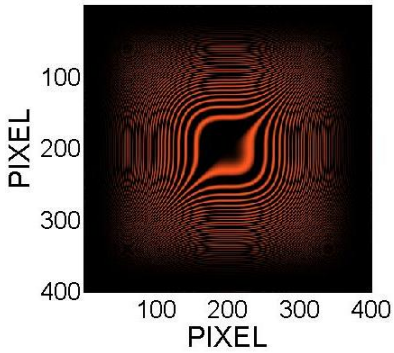
(b). Amplitud de $f_e(x', y')$



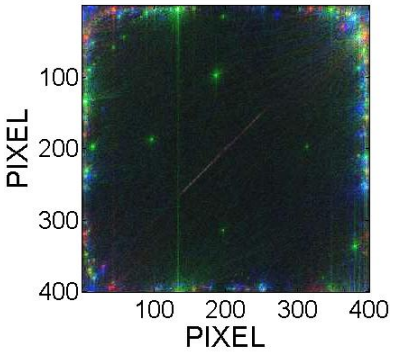
(c). Máscara $a = b = 10$; $p = 2$



(d). Amplitud de $f_e(x', y')$



(e). Máscara $a = b = 0.4$; $p = 3$



(f). Amplitud de $f_e(x', y')$

Figura 4.12. Camuflaje del criptograma multiplicando la entrada con máscaras de simetría circular. Caso III. Llave con CHC $m = 1.5$.

Es importante resaltar que cada criptograma camuflado tiene una apariencia diferente, que en términos de seguridad es un factor a favor. Estos resultados

inéditos, dieron pie para que desarrolláramos, en lenguaje MatLab, un procesador óptico virtual de camuflaje de criptogramas, al que identificamos con el nombre comercial CryptoGOM.CHK. En el Anexo I incluimos aspectos técnicos de este desarrollo computacional y el respectivo manual de usuario.

Capítulo V

5. CONCLUSIONES

Implementamos procesadores ópticos virtuales PEVLC y PEJTC que permiten encriptar imágenes BITMAP de cualquier tipo: Binaria, indexada, de intensidad o RGB. Se estudió la varianza del proceso de desencriptación de la imagen encriptada con la rotación de la llave en coordenadas cartesianas y llave en componentes armónicas circulares (CHC).

Profundizamos el estudio de la varianza de la desencriptación con la rotación de la llave en el PEVLC. Usando llaves con fase en coordenadas cartesianas, tratamos dos casos, sin y con máscara de fase aleatoria multiplicando la imagen a encriptar; encontramos que los resultados sin máscara permiten una tolerancia a la rotación de la llave de $\pm 1°$. En el segundo caso, es menor la tolerancia a la rotación de la llave (hasta $\pm 0.2°$), en su defecto el nivel de encriptación aumenta respecto al caso sin máscara.

El estudio de la varianza utilizando llaves en CHC se hizo con y sin máscara multiplicando la imagen a encriptar. En los resultados obtenidos sin utilizar máscara encontramos una tolerancia a la rotación de la llave de $\pm 60°$, pero con una disminución del nivel de encriptación. En un segundo caso utilizamos como máscara una pupila circular binaria, que generó un mayor nivel de encriptación y una reducción en la tolerancia a la rotación de la llave (hasta $\pm 20°$). Finalmente, utilizamos máscaras determinísticas de estructura periódica, que provocan un efecto de varianza total del proceso de desencriptación con la rotación de la llave. Adicionalmente para el caso III, encontramos como resultado inédito y novedoso, que los criptogramas son determinísticos y con alto nivel de seguridad, que

nosotros hemos denominado criptogramas camuflados, que significa un mecanismo distractor contra los hackers, en la medida que los criptogramas son totalmente diferentes, siendo así un parámetro más de seguridad.

Los criptogramas camuflados, es la nueva estrategia de encriptación de imágenes que dejamos como valor agregado de esta investigación; así mismo, es un valor agregado el desarrollo de un procesador óptico virtual de camuflaje de criptogramas, al que identificamos con el nombre comercial CryptoGOM.CHK.

Del estudio sobre la varianza del procesador PEJTC, encontramos resultados infructuosos, especialmente cuando utilizamos imágenes RGB, razón por la cual, y por limitaciones de tiempo, decidimos no estudiar la varianza a la rotación en este procesador. Queda entonces como perspectiva para trabajos futuros realizar este estudio en profundidad, también queda como perspectiva de una investigación futura, realizar la construcción óptica de procesadores PEVLC y PEJTC usando llaves en CHC.

REFERENCIAS BIBLIOGRÁFICAS

[1] V. Meavilla, «La criptografía clasica,» *SIGMA,* nº 24, pp. 119-141, 2004.

[2] E. A. R. Muñoz, Encriptación y Desencriptación Dinámica de Información por Medios Óptico-Digitales (tesis doctoral), Medellin: Universidad de Antioquia, 2009.

[3] A. Ludia, «Implementacion de tecnicas opticas de seguridad informatica mediante el uso de medios opticos no lineales (tesis maestria, Dir. J. E. Rueda),» Universidad de Pamplona, Pamplona, 2006.

[4] J. Knight, Encyclopedia of espionage, intelligence, and security, Gale Group, 2004.

[5] A. B. Klimov, «información cuántica: ideas y perspectiva,» *Cinvestav,* enero-marzo 2008.

[6] A. Ekert, «introduction to Quantum Computation,» *Springer-Link,* vol. 587, pp. 47-76, 26 Agosto 2002.

[7] G. Z. Bahram Javidi, «Experimental demonstration of the random phase encoding technique for image encryption and security verification,» *Opt. Eng,* vol. 35, nº 9, pp. 2506-2512, 1996.

[8] A. Salazar, J. Rueda y M. Lasprilla, «Encriptación por conjugación de fase en un BSO Utilizando,» *Revista Colombiana de física,* vol. 34, nº 2, 2002.

[9] E. A. R. y. J. F. B. Carlos A. Ríos, «Sistemas ópticos de encriptacion de doble mascara de fase bajo arquitectura 4f,» *Tecno Lógicas,* pp. 77-79, Diciembre 2010.

[10] M. Tebaldi, J. Barrera, N. Bolognini y R. Torroba, «Estudio de las limitaciones de los dispositivos de encriptación óptica multiple,» mayo 2013.

[11] W. C. a. X. Chen, «Optical cryptography topology based on a three-dimensional particle-like distribution and diffractive imaging,» *Optics Express,* vol. 19, nº 10, 2011.

[12] P. R. a. B. Javidi, «Optical image encryption based on input plane and Fourier plane random encoding,» *Opt. Lett,* vol. 20, p. 767–769, 1995.

[13] T. Nomura y B. Javidi, «Optical encryption using a joint transform correlator architecture,» *Optical Engineering,* vol. 39, nº 8, p. 2031–2035, 2000.

[14] X. Tan, O. Matoba, T. Shimura, K. Kuroda y B. Javidi, «Secure optical storage that uses fully

phase encryption,» *Applied Optics,* vol. 39, nº 35, pp. 6689 - 6694, 2000.

[15] B. Javidi y O. Motoba, «Encrypted optical storage with wavelength-key and random phase codes,» *Applied Optics,* vol. 38, nº 32, pp. 6785-6790, 1999.

[16] B. Javidi y O. Motoba, «Secure three-dimensional data transmission and display,» *Applied Optics,* vol. 43, nº 11, pp. 2285-2291, 2004.

[17] B. Javidi, N. Towghi, N. Maghzi y S. Verrall, «Error-reduction techniques and error analysis for fully phase- and amplitude-based encryption,» *Applied Optics,* vol. 39, nº 23, pp. 4117- 4130, 2000.

[18] B. J. a. Z. L. N. Towghi, «Fully phase encrypted image processor,» *J. Opt. Soc. Am. A,* vol. 16, nº 8, pp. 1915-1923, 1999.

[19] L.-C. L. a. C.-J. Cheng, «Optimal key mask design for optical encryption based on joint transform correlator architecture,» *Optics Communications,* nº 258, p. 144–154, 2006.

[20] E. Tajahuerce, O. Matoba, S. Verrall y B. Javidi, «Optoelectronic information encryption with phase-shifting interferometry,» *Applied Optics,* vol. 39, nº 14, pp. 2313 - 2320, 2000.

[21] X. Meng, L. Cai, X. Xu, X. Yang, X. Shen, G. Dong y Y. Wang, «Two-step phase-shifting interferometry and its application in image encryption,» *Optics Letters,* vol. 31, nº 10, pp. 1414 - 1416, 2006.

[22] L. Y. B. Z. Shutian Liu *, «Optical image encryption by cascaded fractional Fourier transforms with random phase filtering,» *Optics Communications,* nº 187, pp. 57-63, 2001.

[23] C.-L. Chen, L.-C. Lin y C.-J. Cheng, «Design and implementation of an optical joint transform encryption system using complex-encoded key mask,» *Optical Engineering,* vol. 47, nº 6, pp. 1-8, 2008.

[24] M. N. Islam, M. S. Alam y M. A. Karim, «Optical security system employing quadrature multiplexing,» *Opt. Eng.,* nº 47(4), abril 17 2008.

[25] M. Islam, M. Karim, M. Alam y M. Asari, «Optical cryptographic system employing multiple reference-based joint transform correlation technique,» *Optical Engineering,* vol. 50, nº (6), 2011.

[26] L. Z. N. Z. Jianhua Wu, «Image encryption based on the multiple-order discrete fractional cosine transform,» *Optics Communications,* nº 283, p. 1720–1725, 2010.

[27] M. Tebaldi, W. D. Furlan, R. Torroba y N. Bolognini1, «Optical-data storage-readout technique based on fractal encrypting masks,» *Optics Letters,* vol. 34, nº 3, pp. 316-318, 2009.

[28] J. F. Barrera, M. Tebaldi, D. Amaya, W. D. Furlan, J. A. Monsoriu, N. Bolognini y R. Torroba, «Multiplexing of encrypted data using fractal masks,» *Optics Letters,* vol. 37, nº 14, pp. 2895-2897, 2012.

[29] A. S. Narendra Singh, «Optical image encryption using improper Hartley transforms and chaos,» *Optik,* nº 121, p. 918–925, 2010.

[30] J. Barrera, M. Tebaldi, D. Amaya, W. Furlan, J. Monsoriu, N. Bolognini y R. Torroba, «Experimental multiplexing of encrypted movies using a JTC architecture,» *Optics Express,* vol. 20, nº 4, 2012.

[31] M. Joshi, C. Shakher y K. Singh, «Image encryption using radial Hilbert transform filter bank as an additional key in the modified double random fractional Fourier encoding architecture,» *Optics and Lasers in Engineering,* nº 48, p. 605–615, 2010.

[32] Q. Wang, Q. Guo y J. Zhou, «Multiple-image encryption using polarized light encoding and the optical interference principle in the Fresnel-transform domain,» *Applied Optics,* vol. 52, nº 36, pp. 8854-8863, 2013.

[33] X. Li y D. Zhao, «Optical color image encryption with redefined fractional Hartley transform,» *Optik,* nº 121, p. 673–677, 2010.

[34] X. Ding y G. Chen, «Optical color image encryption using position multiplexing technique based on phase truncation operation,» *Optics y Laser Technology,* nº 57, p. 110–118, 2014.

[35] C.-H. Niu, X.-L. Wang y X.-H. Mao, «Multiple-image hiding based on interference principle,» *Springer,* p. 91–99, 2011.

[36] C. Lin, X. Shen y W. Hu, «Information encryption and compression based on random polarization modulation in a joint transform correlator scheme under vector beam illumination,» *Springer ,* pp. 1-7, 2015.

[37] I. Muniraj, B. Kim y B.-G. Lee, «Encryption and volumetric 3D object reconstruction using multispectral computational integral imaging,» *Applied Optics,* vol. 53, nº 27, pp. G25-G32, 2014.

[38] X. Li, C. Tang, X. Zhu, B. Li, L. Wang y X. Yan, «Image/video encryption using single shot digital holography,» *Optics Communications,* nº 342, p. 218–223, 2015.

[39] A. Yadav, S. Vashisth, H. Singh y K. Singh, «A phase-image watermarking scheme in gyrator domain using devil's vortex Fresnel lens as a phase mask,» *Optics Communications,* nº 344, p. 172–180, 2015.

[40] J. E. Rueda, «Encryption using circular harmonic key,» *DYNA,* vol. 82, nº 190, pp. 1-10, 2015.

[41] W. Zamrani, E. Ahouzi, A. Lizana, J. Campo y M. Yzuel, «Optical image encryption technique based on deterministic phase masks,» *Optical Engineering,* vol. 55, nº 10, pp. 1-9, 2016.

[42] J. W. Goodman, introduction to Fourier Optics, The McGraw hill, 1996.

[43] F. M. Grilmaldi, «físico mathesis de Lumine, coloribus, et iride, aliisque annexis libri duo (Bologna("Bonomia")),» pp. 1-11, 1665.

[44] V. Lugt, «Signal detection by complex spatial filtering,» *IEEE transactions on Information,* vol. 10, p. 139, abril 1964.

[45] M. N. Islam, M. S. Alam y M. A. Karim, «Optical security system employing quadrature multiplexing,» *Optical Engineering,* vol. 47, nº 4, pp. 1-5, 2008.

[46] Q. G. M. Huynh-Thu, «Âmbito de validade do PPSNR na avaliação da qualidade de imagem / vídeo,» *Electronics Letters,* vol. 44, nº 13, 2008.

[47] E. W. Hansen, «Theory of circular harmonic image reconstruction,» *J. Opt. Soc. Am.,* vol. 71, nº 3, pp. 304-308, 1981.

[48] E. W. Hansen, «Circular harmonic image reconstruction: experiments,» *Applied Optics,* vol. 20, nº 13, pp. 2266-2274, 1981.

[49] J. Hofer, «Optical reconstruction from projections via deconvolution,» *Optics Communications,* vol. 29, nº 1, pp. 22 - 26, 1979.

[50] O. Gualdron y H. Arsenault, «Phase dirived circular harmonic filter,» *Optics Communications,* vol. 104, pp. 32-34, 1993.

[51] C. S. W. a. J. W. Goodman, «A Technique for Optically Convolving Two Functions,» *Applied Optics,* vol. 5, pp. 1248-1249, 1966.

[52] M. N. Islam, M. S. Alam y M. A. Karim, «Optical security system employing quadrature multiplexing,» *Optica Engineering,* vol. 47, nº (4), abril 17 2008.

[53] J. A. V. Valencia, Métodos Optimizados de Multiplexado y Encriptación Óptico-Digital, Medellin, 2014.

[54] D. Sosa, M. Tebaldi, S. Horrillo, E. Pérez-Cabré y M. Millán, «Multiplexado en color para esquemas ópticos de encriptación,» de *3ras Jornadas ITE*, abril 2015.

[55] O. M. a. B. Javidi, «Encrypted optical storage with angular multiplexing,» *Applied Optics,* vol. 38, nº 35, pp. 7288-7293, 1999.

[56] H. Yuang-Neng, H. Arsenault y G. April, «Rotation-invariant digital pattern recognition using circular harmonic expansion,» *Applied Optics,* pp. 1-2, 15 noviembre 1982.

[57] J. Wu, Z. Xie, Z. Liu, W. Liu, Y. Zhang y S. Liu, «Multiple-image encryption based on computational ghost imaging,» *Elsevier,* vol. 359, pp. 38-43, 2015.

ANEXO I. MANUAL TÉCNICO Y DE USUARIO DE PROCESADOR ÓPTICO VIRTUAL DE CAMUFLAJE DE CRIPTOGRAMAS CRYPTOGOM.CHK

Nota: la expresión explicita que genera la máscara $\boldsymbol{h^{-1}(ax^p, by^p)}$ y el código de la herramienta CRYPTOGOM.CHK no se incluyen en el documento, por motivos del registro comercial del producto y publicación del tema.

MANUAL TÉCNICO:

CRYPTOGOM.CHK es un procesador óptico virtual de encriptación PEVLC que permite generar criptogramas camuflados como una imagen determinística. Las imágenes de entrada son BITMAP de cualquier tipo: Binaria, de intensidad, indexada o RGB, y de cualquier formato que soporte MatLb (bmp, jpg, tif, etc) Esta herramienta está desarrollada en lenguaje MATLAB. La arquitectura es simétrica, es decir de clave privada (se encripta y desencripta con la misma llave). Se utilizó el entorno visual interfaz de usuario GUI_MATLAB (ver **Figura A. 1**). El diseño de la interfaz gráfica consta de tres partes: menu principal (ver **Figura A. 2**), ventana de Encriptación (ver **Figura A. 3**) y ventana de Desencriptación (ver **Figura A. 4**).

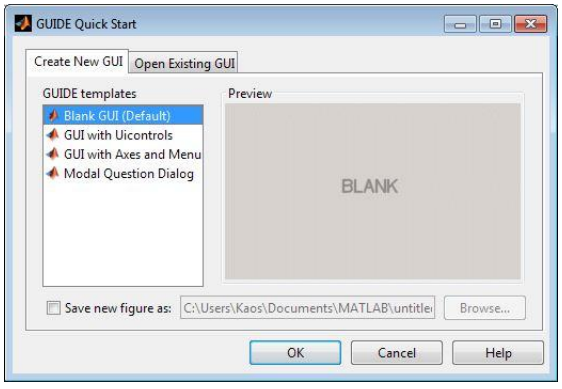


Figura A. 1. Ventana de inicio de construcción de GUI.

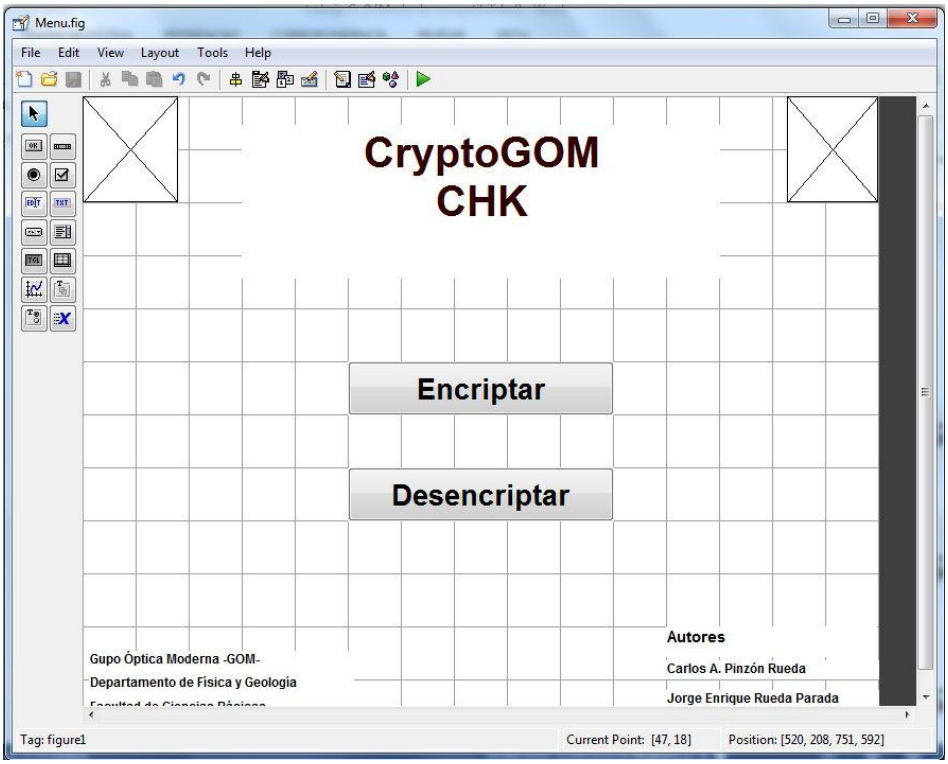


Figura A. 2. Entorno de diseño; menú CryptoGOM.CHK

Los botones "Encriptar" y "Desencriptar" enlazan las ventanas Encriptación y Desencriptacion. Para el proceso de encriptación se realizó una interfaz gráfica (ver **Figura A. 3**).

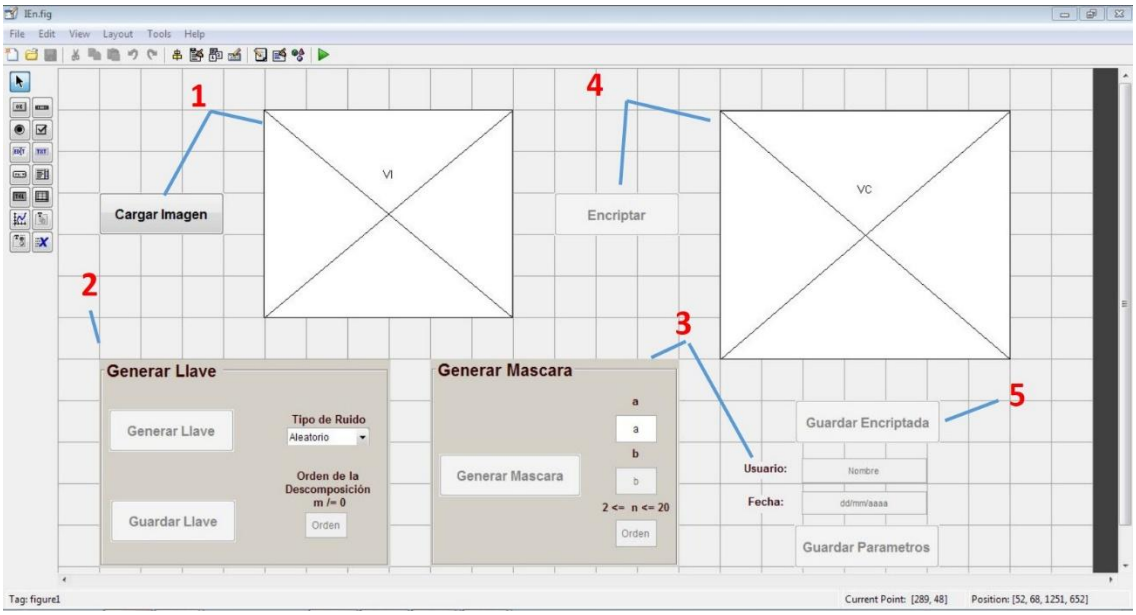


Figura A. 3. Entorno de diseño; ventana de encriptación CryptoGOM.CHK.

Descripción de la ventana encriptación:

1. Botón *"Cargar Imagen"*: Permite llamar y visualizar la imagen que se desea encriptar.
2. panel *"Generar Llave"*: genera la llave de fase, teniendo en cuenta dos parámetros de entrada; 1. *"tipo de ruido"* (tipo aleatorio o Speckel); 2. orden de la descomposición en armónicos $m \neq 0$, valor que el usuario ingresa por teclado donde; el botón *"Guardar Llave"* permite guardar la llave como una archivo *.mat.
3. Panel *"Generar Máscara"*: requiere de tres parámetros de entrada, que nos permitirán crear la máscara determinística de tipo $\boldsymbol{h^{-1}(ax^p, by^p)}$; los parámetros a y b debes ser diferentes de cero ($\boldsymbol{a \neq 0}$; $\boldsymbol{b \neq 0}$); el parámetro p debe ser ($\boldsymbol{2 \leq p \geq 20}$); el botón "Generar Mascara" genera una máscara determinística teniendo en cuenta los tres parámetros ingresados; el botón *"Guardar Parámetros"* almacena el nombre de usuario, la fecha y los parámetros a, b y p en una archivo *.txt.

4. El botón *"Encriptar"* permite encriptar la imagen y visualizarla.
5. El botón *"Guardar Encriptada"* permite guardar dos archivos *.mat de la imagen encriptada; el primero corresponde al módulo de la imagen encriptada; el segundo corresponde a la fase de la imagen encriptada.

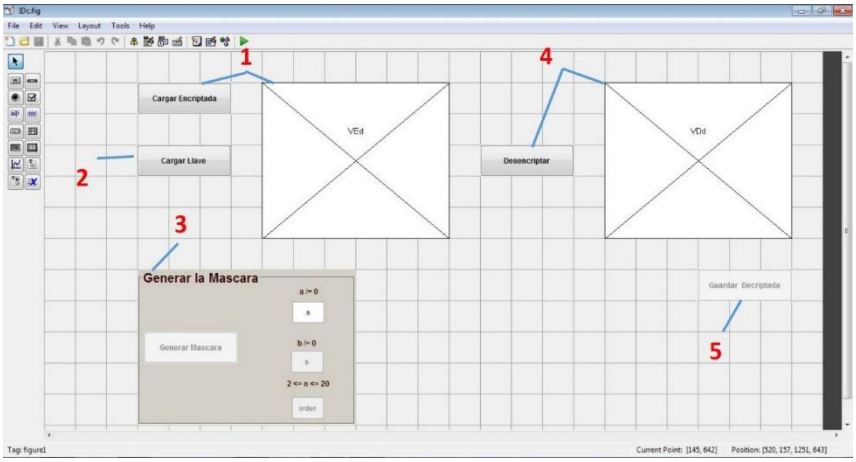


Figura A. 4. Entorno de diseño; ventana de desencriptación CryptoGOM.CHK.

El segundo botón del menú *"Encriptar"* nos lleva a una segunda ventana, donde se realizará el proceso de desencriptación de la información. A continuación se explica la funcionabilidad de cada una de las partes de la interface (véase **Figura A. 4**);

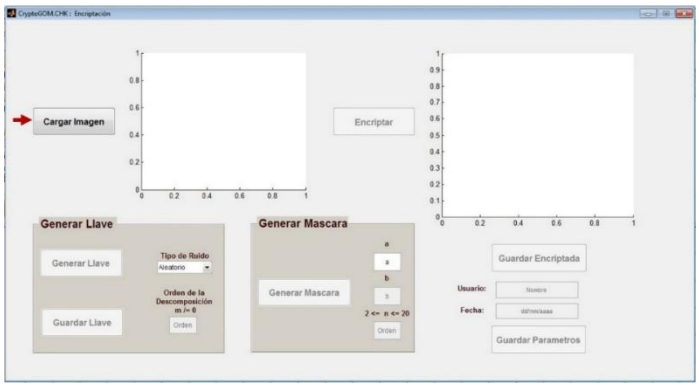
1. Botón *"Cargar Encriptada":* permite llamar los archivos de la imagen encriptada (modulo.mat y fase.mat) y visualizar el módulo.
2. Botón *"Cargar Llave":* permite cargar el archivo *.mat de la llave.
3. Panel "Generar Máscara": permite generar la máscara determinística ingresando los mismos parámetros usados en el proceso de encriptación ($a, b\ y\ p$).
4. Botón *"Desencriptar":* permite desencriptar la imagen encriptada y visualizarla.

5. Botón *"Guardar Desencriptada":* permite guardar la imagen desencriptada en cualquier formato de imagen.

MANUAL DE USUARIO

CryptoGOM.CHK: es la interfaz principal (**Figura A. 5(a)**) que contiene el menú Encriptar y Desencriptar. Se acciona el botón Encriptar y emerge la ventana *"CryptoGOM.CHK: Encriptación" (***Figura A. 5*(b)**). Se acciona el botón Desencriptar y emerge la ventana *"CryptoGOM.CHK: Desencriptación"(***Figura A. 13*(b)**).

(a). ventana menú de CryptoGOM.CHK



(b). ventana CryptoGOM.CHK: Encriptación

Figura A. 5. Encriptación.

En la ventana de *"CryptoGOM.CHK: Encriptación"* el usuario primero debe carga la imagen que se desea encriptar, dando click en el botón *"Cargar Imagen"* donde emergerá una ventana la cual le permitirá buscar y seleccionar la imagen que desea encriptar (ver **Figura A. 6**).

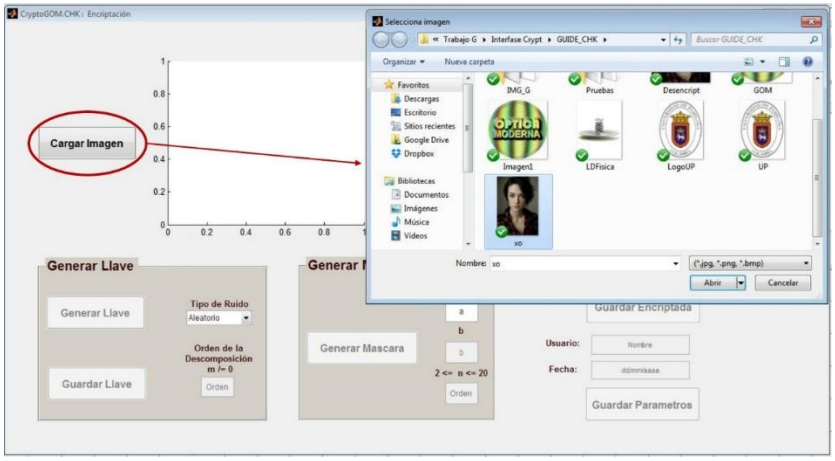


Figura A. 6. Ventana encriptación; proceso cargar imagen.

El tipo de formato de la imagen que recibe *CryptoGOM.CHK* es de tipo *BITMAP (jpg, png, bmp, etc)* luego de seleccionar la imagen esta se visualiza en el primer cuadrante imagen.

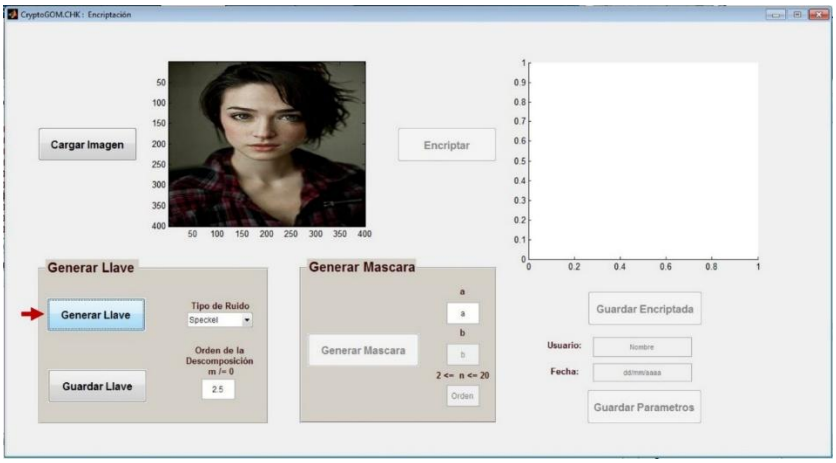


Figura A. 7. Ventana encriptación; proceso generar llave.

Despues de seleccionar la imagen, en el panel “Generar Llave” el usuario debe ingresar los párametros (tipo de ruido y orden de la descomposicion), luego accionar el boton *“Generar Llave”* (**Figura A. 7**). El boton *“Guardar Llave”* permite al usuario elegir la direccion en donde desee guardar la llave, tal como se muestra en la **Figura A. 8**.

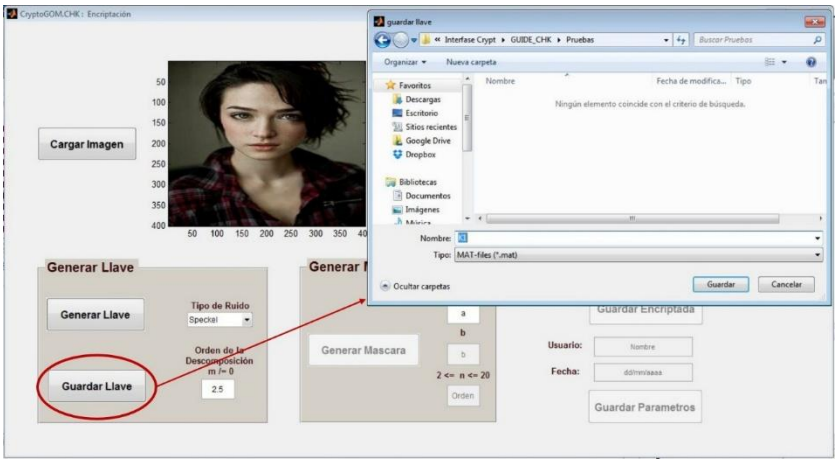


Figura A. 8. Ventana encriptación; proceso guardar llave.

Panel *"Generar Máscara"* : el usuario debe ingresar los parametros de entrada (a,b y p), y se acciona el botón *"Generar Máscara"* (ver **Figura A. 9**).

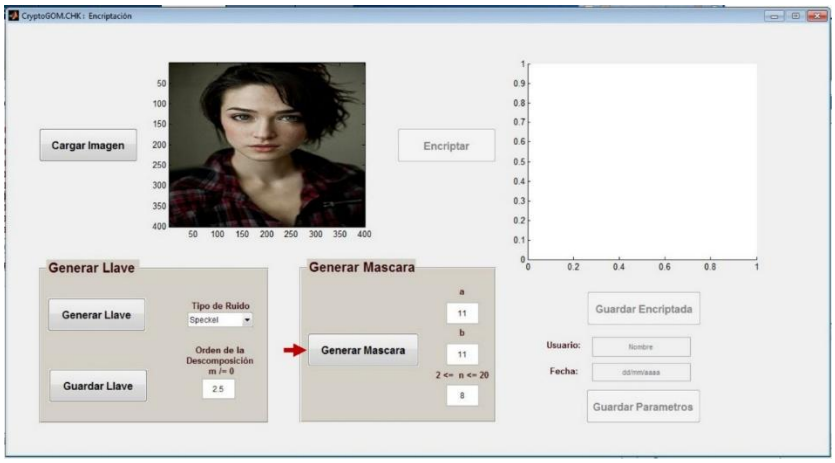


Figura A. 9. Ventana encriptación; proceso generar mascara.

Habiendo generando la máscara y la llave, se procede a encriptar la imagen accionando el botón *"Encriptar"* (ver **Figura A. 10**).

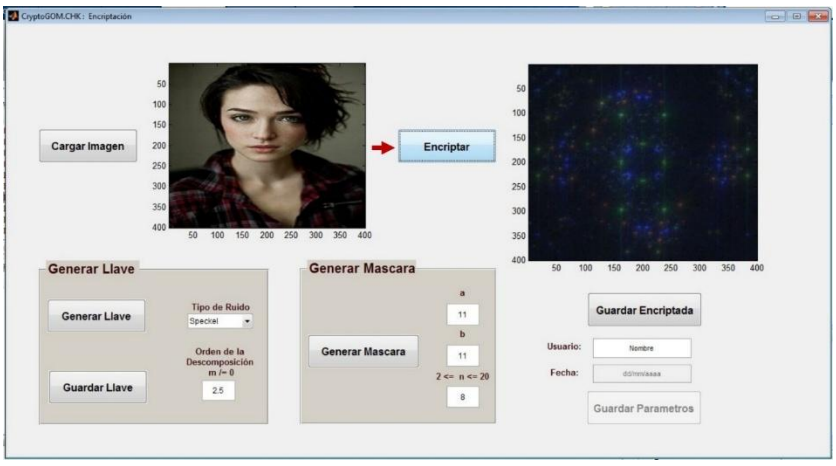
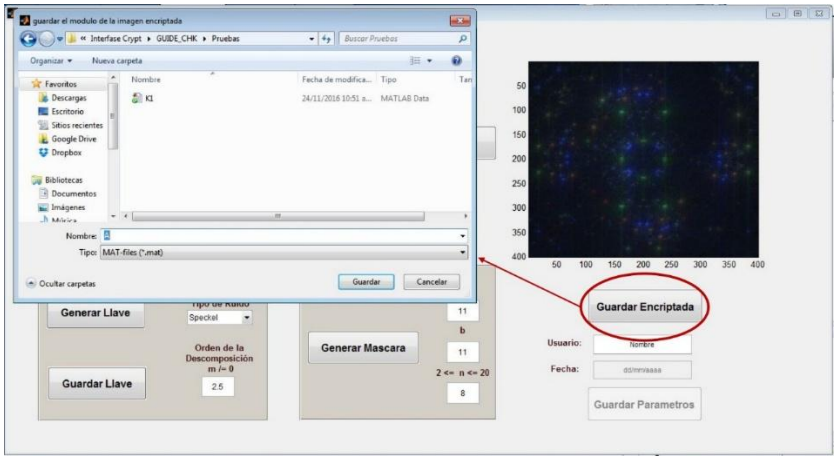
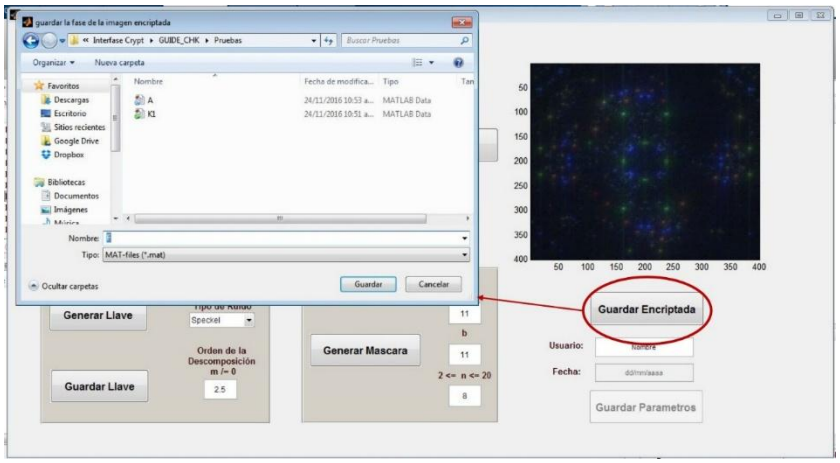


Figura A. 10. Ventana encriptación; proceso encriptación.

Especificado los parametros usados en el proceso de encriptación y la misma imagen encriptada, el usuario puede proceder a almacenar la imagen encriptada y los parametros de encriptación. El boton *"Guardar Encriptada"* permite al usuario guardar dos archivos *.mat correspondientes al modulo y la fase de la imagen encriptada de forma secuencial (ver **Figura A. 11**).



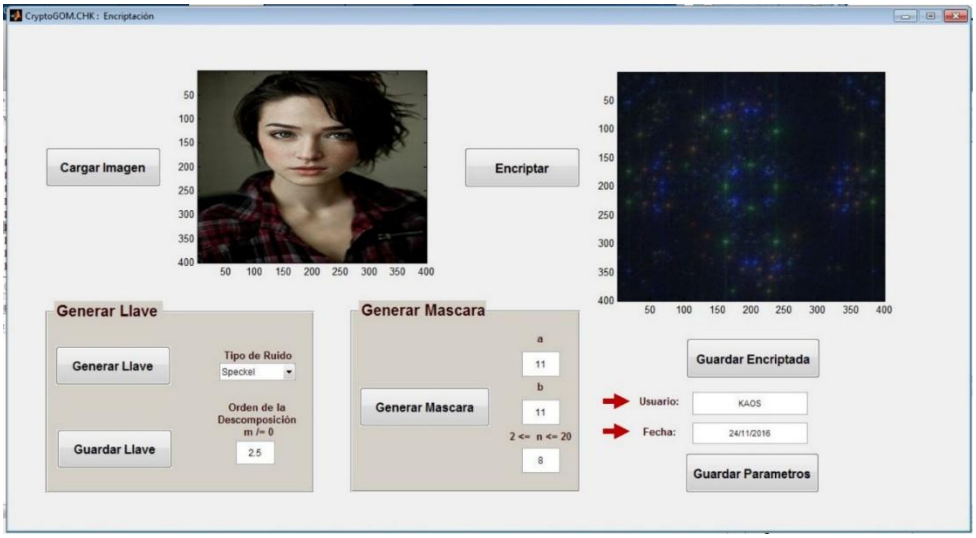
(a). Guardar módulo de la imagen encriptada.



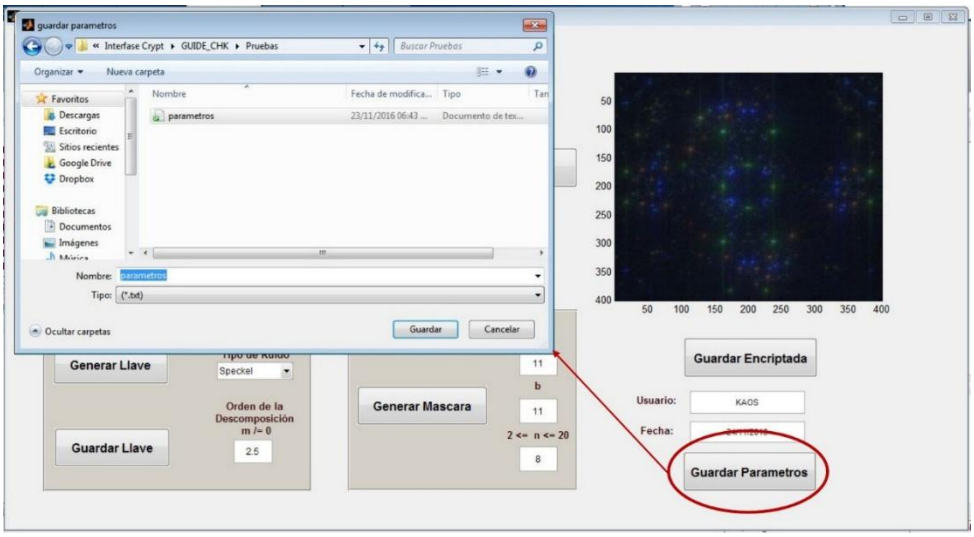
(b). Guardar fase de la imagen encriptada.

Figura A. 11. Ventana encriptación; proceso guardar imagen encriptada.

El usuario podrá ingresar su nombre, fecha y los parámetros de la máscara que serán almacenados en un archivo .txt accionando el botón *"Guardar Parámetros"*, (ver **Figura A. 12**).



(a). ingreso de nombre de usuario y fecha.



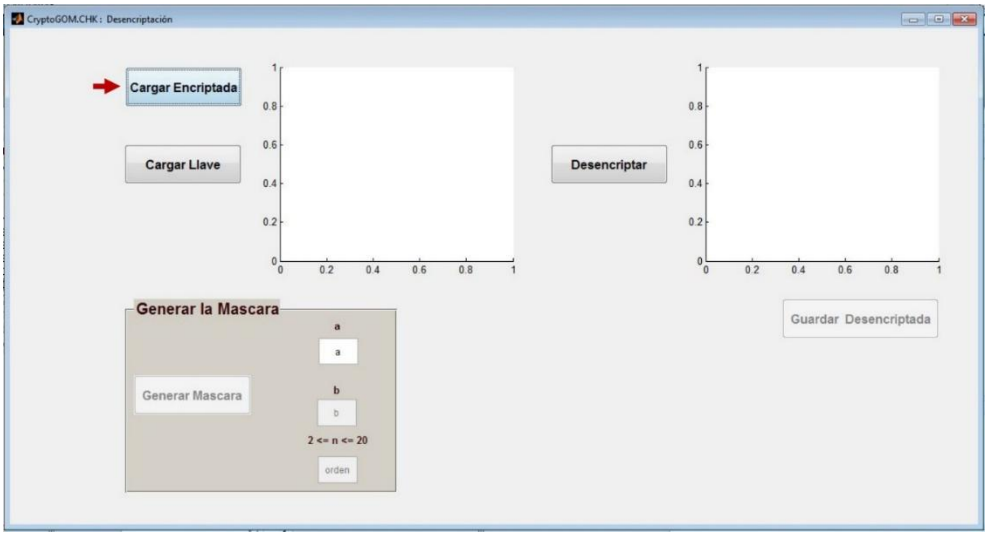
(b). guardar parámetros

Figura A. 12. Ventana encriptación; proceso guardar parámetros.

Desencriptación de la imagen:

En el menú principal se acciona el botón *"Desencriptar"* que abre la ventana *"CryptoGOM.CHK: Desencriptacion"* (ver **Figura A. 13**)

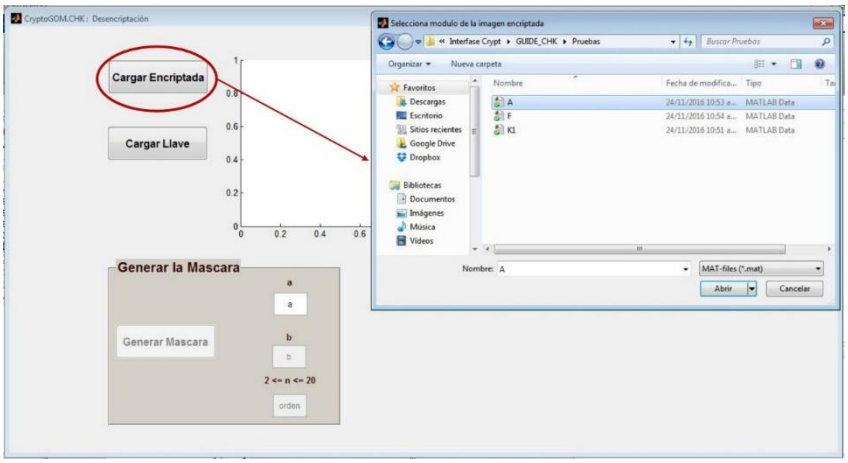
(a). Ventana menú CryptoGOM.CHK.



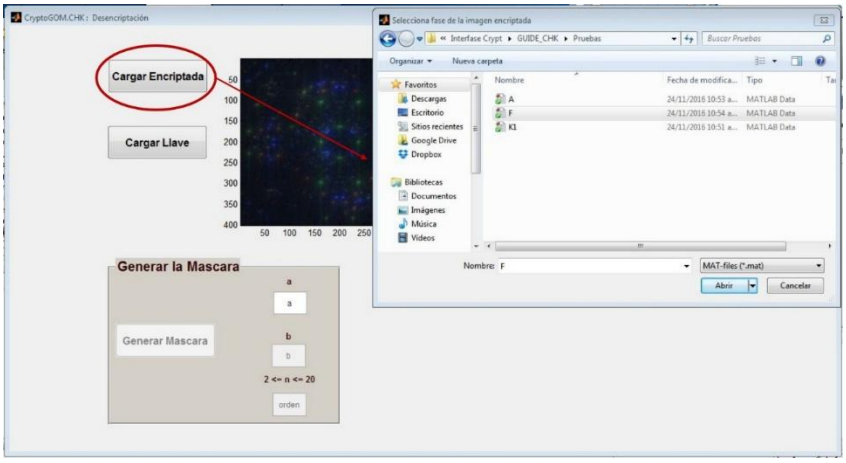
(b). Ventana CryptoGOM.CHK: Desencriptación.

Figura A. 13. Desencriptación.

El botón *"Cargar Encriptada"* permite seleccionar de manera secuencial los archivos modulo y fase de la imagen encriptada (ver **Figura A. 14**).



(a). Cargar módulo de la imagen encriptada.



(b). Cargar fase de la imagen encriptada.

Figura A. 14. Cargar imagen encriptada.

Accionando el botón *"Cargar Llave"* el usuario podrá seleccionar la llave de desencriptación (ver **Figura A. 15**).

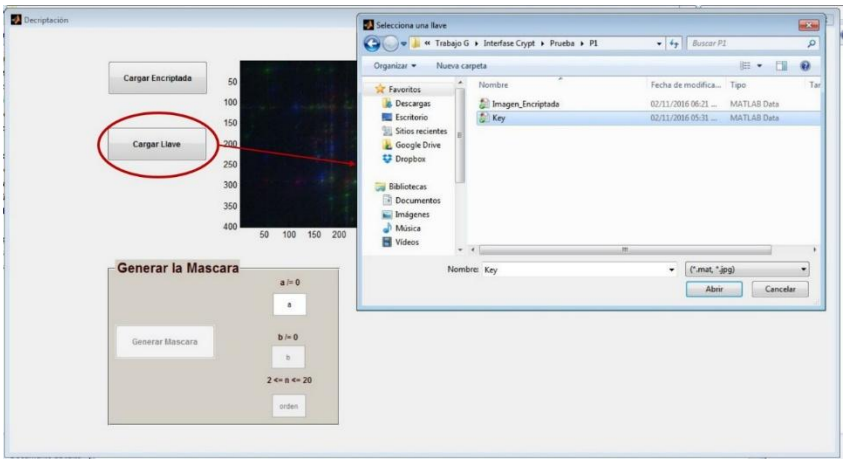


Figura A. 15. Ventana desencriptación; proceso cargar llave.

Después de cargar la llave, el usuario debe ingresar los parámetros de la máscara (a, b y p), y accionar el botón *"Generar Mascara"* (ver **Figura A. 16**).

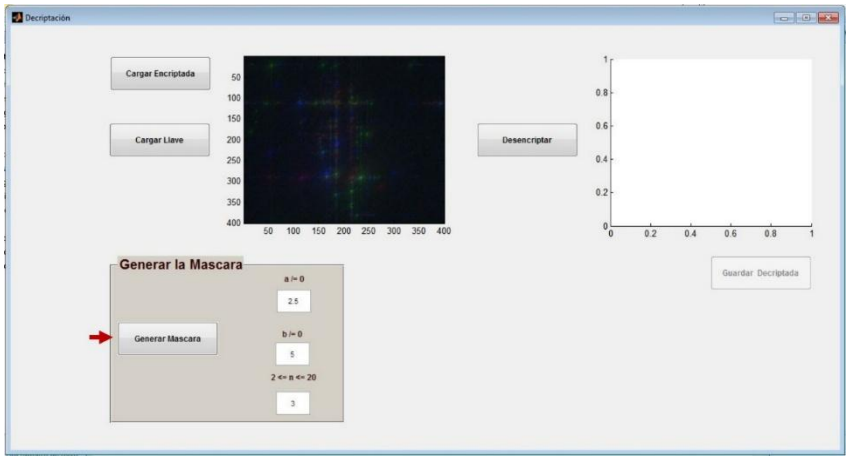


Figura A. 16. Ventana desencriptación; proceso generar mascara.

Habiendo cargado la llave y generado la máscara, el usuraio podrá desencriptar y visializar la imagen con el botón *"Desencriptar"* (ver **Figura A. 17**).

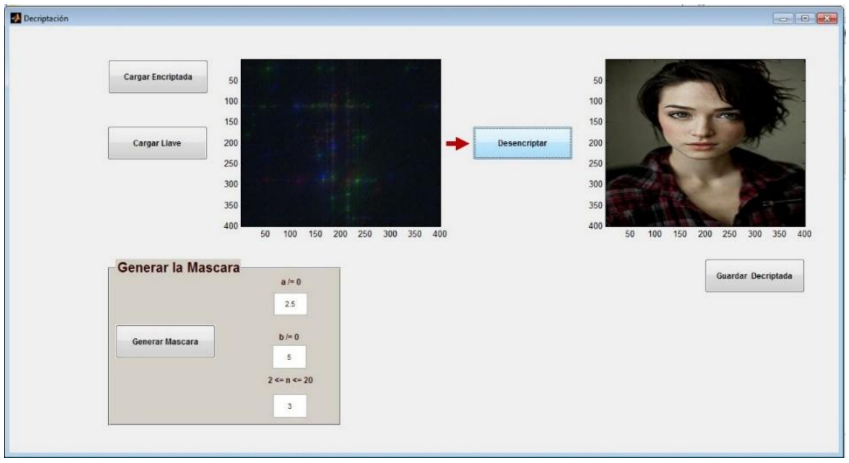


Figura A. 17. Ventana desencriptación; proceso desencriptación.

Por último el usuario podrá guardar la imagen desencriptada usando el botón *"Guardar Desencriptada"* (ver **Figura A. 18**).

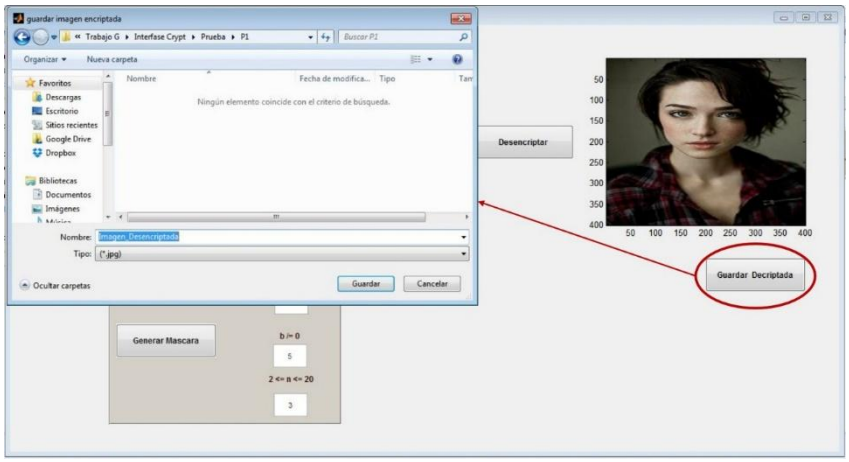


Figura A. 18. Ventana desencriptación; proceso guardar imagen desencriptada.