



OPTOEL 2021  
Evento online  
30 de junio, 1 y 2 de julio 2021

# XII REUNIÓN NACIONAL DE OPTOLECTRÓNICA



*Organizado por:*

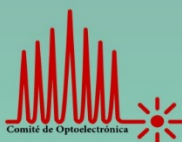


**Universidad  
Zaragoza**



UNIVERSITAT ID VALÈNCIA

*Con colaboración de:*



**Editado por:**

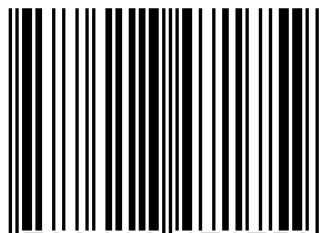
María Teresa Flores Arias

Patricia Fraga María

**Publicado por:**

Comité de Optoelectrónica

ISBN 978-84-18471-75-9



9 788418 471759 >

# Efecto de la codificación en el número de errores introducidos en un HDSS

## Codification effect in the error number introduced in a HDSS

Fco. Javier MARTÍNEZ-GUARDIOLA<sup>(1,2)</sup>, Andrés MÁRQUEZ<sup>(1,2)</sup>, Eva CALZADO<sup>(1,2)</sup>, Sergi GALLEGO<sup>(1,2)</sup>, Daniel PUERTO<sup>(1,2)</sup>, Marta MORALES-VIDAL<sup>(2)</sup>, Inmaculada PASCUAL<sup>(2,3)</sup>, Augusto BELÉNDEZ<sup>(1,2)</sup>

1. Departamento de Física, Ingeniería de Sistemas y Teoría de la señal, Universidad de Alicante, Crta. San Vicent del Raspeig s/n.
2. Instituto Universitario de Física Aplicada a las Ciencias y las Tecnologías, Universidad de Alicante, Crta. San Vicent del Raspeig s/n
3. Departamento de Óptica, Farmacología y Anatomía, Universidad de Alicante, Crta. San Vicent del Raspeig s/n.

Persona de contacto: F. J. Martínez-Guardiola ([fj.martinez@ua.es](mailto:fj.martinez@ua.es)).

### RESUMEN:

En este trabajo se presentan resultados preliminares sobre el efecto de la codificación de los datos digitales introducidos en un sistema de almacenamiento holográfico de información. Para ello se introducen una serie de fuentes de ruido que afectan a las imágenes que usamos como entrada de datos. Estas fuentes de ruido son las que se observan principalmente en el montaje real, por lo que se trata de imitar la degradación del sistema para poder tener un modelo que nos ayude a identificar los parámetros más disruptivos en el tratamiento óptico de la señal y cómo influye la introducción de una codificación en dicha imagen.

**Palabras clave:** Holografía, Almacenamiento Digital, Dispositivos de Cristal Líquido, PA-LCoS, Modelado, SNR

### ABSTRACT:

In this work we present preliminary results about the codification effect over the digital data introduced in a Holographic Data Storage System (HDSS). In this sense, we introduce a series of noise sources that affects to the images used as data in the system input. These noise sources are related with the different degradations that we observed in the real setup, we are trying to model the system degradation in order to obtain a computational model that helps us to indentify the more disruptive parameters in the optical signal processing and how affects the introduction of a codification in the input image.

**Key words:** Holography, Data Storage, Liquid-Crystal devices, PA-LCoS, Modeling, SNR

## 1.- Introducción

Desde la primera aparición de los láseres en los años 60 [1] se ha desarrollado un interés en la industria por el almacenamiento holográfico de información (HDS, de sus siglas en inglés Holographic Data Storage). Aunque es una tecnología muy prometedora,

lleva asociado una importante cantidad de desafíos científicos e ingenieriles que hay que tratar de superar y que suponen un campo de estudio e investigación en sí mismo. Esta tecnología posibilita un verdadero almacenamiento en las 3 dimensiones del material, y además permite la búsqueda asociativa de información [2].

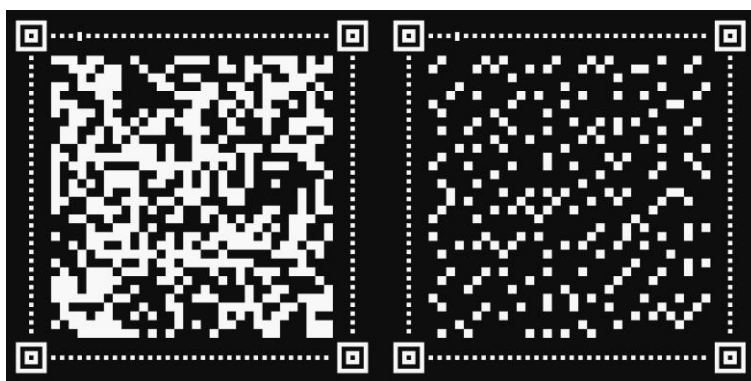


Fig. 1: Imágenes utilizadas como entrada de datos, sin codificación (izquierda) y con codificación (derecha).

En su corta historia han aparecido algunas plataformas de desarrollo y prototipos comerciales, cuya principal aplicación sería el almacenamiento de larga duración [3-6].

En este trabajo nos vamos a centrar en el tratamiento de los datos y la degradación que sufren estos en su paso a través del sistema de almacenamiento de datos en su conjunto. También queremos estudiar la influencia de una posible codificación de los datos de entrada en la afectación sufrida por la entrada en las distintas fuentes de ruido identificadas.

En la Figura 1 podemos ver un ejemplo de las imágenes originales que se introducen en el sistema tanto codificadas como sin codificación. En el siguiente apartado explicaremos cómo se generan estas imágenes y sus diferencias.

## 2.- Generación de imágenes

En la generación de imágenes debemos tener en cuenta dos parámetros básicos, el tamaño del bit y si usamos algún tipo de codificación o no.

El tamaño del bit está relacionado con la resolución de nuestro dispositivo para generar las páginas de datos, en nuestro caso utilizamos un dispositivo de cristal líquido reflectivo, concretamente un dispositivo PLUTO de la compañía HOLOEYE. En trabajos anteriores hemos visto como tenemos la limitación por el filtrado que se produce en el montaje experimental [7,8], por lo que siempre vamos a utilizar más de un píxel

para representar un bit de información. Por lo que como vemos en la Figura 1, los bits de información son de un tamaño mayor que un sólo píxel, en el caso del ejemplo de la Figura 1 se ha utilizado un tamaño del bit de 16x16 píxeles para tener una mejor representación.

Una vez definido el tamaño del bit tenemos que optar por codificar la información o no. En el caso de las imágenes sin codificar se ha optado por generar páginas con bit aleatorios de tamaño 512x512 píxeles (el número de bits de información dependerá del tamaño del bit en píxeles). Estas páginas aleatorias se han generado de forma que haya el mismo número de píxeles a uno (blanco) como píxeles a cero (negro), de esta forma controlamos que la energía disponible para cada página es constante y podemos utilizar siempre la misma relación de haces necesaria para la correcta impresión del material de registro [9]. Es de destacar, que en este tipo de imágenes de 512x512 píxeles, si utilizamos un tamaño de bit de 2x2, tendríamos  $256 \times 256 = 65536$  bits de información, o lo que es lo mismo, y que nos servirá para comparar con las páginas codificadas, 8192 bytes de información.

En el caso de las imágenes codificadas se ha optado por probar un código de peso constante, o lo que es lo mismo cada símbolo estará definido por un patrón de  $N \times N$  píxeles de los cuales se encontraran activos (definidos como UNO) un número constante de píxeles. De esta forma, aunque como se ve en Figura 1 el número de píxeles a uno es infe-

rior al número de píxeles a cero, siempre tendremos un número de píxeles a uno igual entre distintas páginas de datos con informaciones distintas, lo que nos obliga a cambiar la relación de haces con respecto a la página sin codificar, pero no entre distintas páginas codificadas de la misma forma.

Se ha seguido la codificación descrita por Gu et al [10]. En esta codificación establece una serie de reglas:

- Símbolo de peso constante de 4x4 bits donde sólo 3 bits estarán activos. Esto nos llevaría a 3360 posibles símbolos. Esto nos fuerza a que la distancia Hamming mínima de un símbolo con otro diferente será de 2.
- Alrededor de un '0' no podrá haber más de 2 píxeles en estado '1'. Esto nos deja con sólo 276 símbolos de los 3360 anteriores.
- Eliminamos aquellos símbolos que tienen más símbolos del conjunto a distancia de Hamming de 2 de sí mismos (4 símbolos menos, 272 en total)
- Volvemos a calcular la distancia de Hamming y volvemos a eliminar los símbolos que tienen distancia de Hamming de 2 la mayoría de las veces. (16 símbolos menos, 256 en total)
- En este punto ya tenemos un conjunto de símbolos, con código de 4x4 bits y que codifican una palabra de 8 bits (256 símbolos).

En este momento tenemos matrices de 4x4 bits que representan palabras de 8 bits, como se muestra en la figura 2. En esta figura se presentan los 8 primeros elementos del código generado y el número que representan. De esta forma tenemos codificado con cada símbolo un byte. Dado que tenemos 256 posibles símbolos con ellos podremos codificar el código ascii de un texto. En nuestro caso hemos elegido un fichero de texto plano que contiene la obra de Miguel de Cervantes "Don Quijote" descargada de la librería de libros electrónicos libres mantenida por el Proyecto Gutenberg (Project Gutenberg)

[11]. Este fichero de texto será el que nos sirva para generar la página de datos codificada como la mostrada en la figura 1 (derecha).

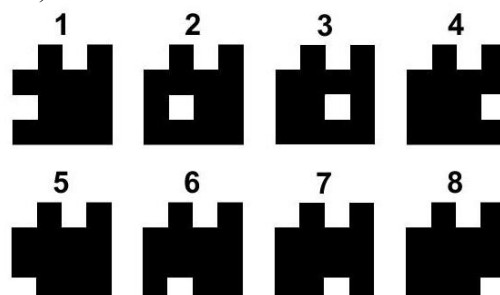


Fig.2 Tabla de codificación de los primeros 8 elementos

La primera consideración que podemos hacer es en relación con la densidad de datos que somos capaces de almacenar en las dos situaciones, como hemos visto anteriormente, sin codificar, 8 bits representaban un byte, en el caso de páginas con codificación tenemos que un byte ocupará 16 bits en la imagen, con lo que la densidad de datos será la mitad. Esto será importante a la hora de comparar los errores detectados ya que en un caso será un símbolo completo leído incorrectamente y en otro caso será un solo bit el que ha fallado. Tenemos también la posibilidad de contar los errores de la página con codificación contando bit a bit para poder hacer comparaciones entre los dos tipos de páginas de datos.

### 3. Modelado de la degradación

Una vez presentado cómo se generan las imágenes con las que alimentaremos nuestro sistema, vamos a presentar las distintas perturbaciones que mejor representan la degradación que sufre la imagen en el paso a través de nuestro HDSS.

La imagen original es captada, a la salida de todo el sistema, por una cámara PCO-1600 con sensor CCD refrigerado y con un rango dinámico de 14 bits. Esto nos lleva a tener la posibilidad de 16384 niveles de gris, con lo que la simulación la haremos con este rango. En trabajos anteriores [9], se observa una dispersión gaussiana de los unos y los ceros, quedando los histogramas de estos datos

solapados en algunos puntos. Este solapamiento es la principal fuente de los errores detectados.

En nuestro modelo podemos seleccionar el valor medio para los píxeles oscuros (ceros) y píxeles claros (unos) así como la desviación típica de estos. En una imagen recuperada real, estos valores dependerán de la intensidad seleccionada en el láser, así como de la configuración de la cámara utilizada para recuperar la información.

Variando estos valores del valor medio y la desviación típica de los píxeles claros y oscuros podemos emular los histogramas que obtenemos experimentalmente en nuestro HDSS [9]. Esto nos produce una imagen dispersada, donde los valores claros y oscuros ya no responden a un único valor sino que se dispersan siguiendo una distribución gaussiana. En la figura 3 mostramos un histograma de las imágenes generadas para este análisis, en ella se puede observar que se ajustan a los datos reales obtenidos en los trabajos citados [9].

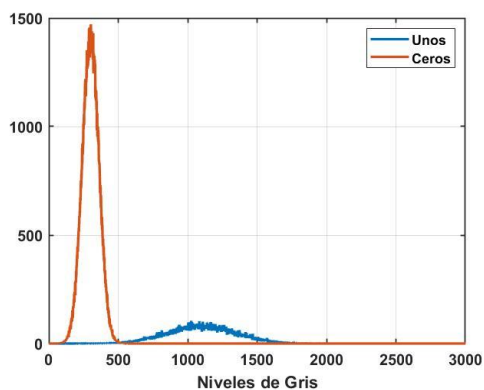


Fig.3: Histograma de una imagen generada para el testeo de estos parámetros. Corresponde a una imagen codificada con tamaño de bit de 2x2 píxeles.

Variando la forma de este histograma variamos el contraste con el que se observa la imagen a la salida. Los niveles de gris pueden variar entre 0 y 16384 ya que la cámara utilizada tiene 14 bits de rango dinámico, aunque en la figura 3 sólo se muestran los niveles de gris hasta 3000 por claridad.

La siguiente fuente de distorsión que podemos variar es la causada por una iluminación no uniforme. Debido a la naturaleza de nuestro sistema experimental tenemos cierto efecto de difracción producido por el paso del haz a través de los distintos elementos ópticos. Esto se traduce en la aparición de una iluminación no uniforme en forma de halos concéntricos.

En la figura 4 se muestra la iluminación no uniforme introducida en las imágenes a analizar. En nuestro sistema podemos combinar esta imagen con la imagen una vez se han dispersado los píxeles. Con respecto a la iluminación no uniforme lo que se varía principalmente es la intensidad de esta iluminación, de manera que será más o menos notable en la imagen final.

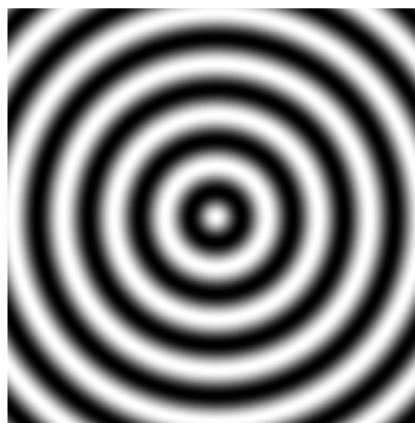


Fig.4: Patrón de iluminación no uniforme introducido en las imágenes para simular la falta de homogeneidad en la iluminación.

La última fuente de error que consideramos es la introducida por la apertura finita del sistema óptico. Lo que simulamos es un filtro paso bajo, filtramos con distintos diámetros considerando un filtro de borde abrupto (Filtro Paso Bajo Ideal, FPB).

Por último, para realizar el conteo del número de errores hay que realizar una binarización de la imagen deteriorada que está en niveles de gris, el tratamiento que hacemos debemos mantenerlo al mínimo con lo que lo único que se hará será binarizar la imagen para comparar con la original. El cálculo del umbral para la imagen se realiza siguiendo el método de Otsu [12], este es un método

clásico utilizado ampliamente en tratamiento digital de imágenes.

#### 4. Resultados

Para mostrar los datos que es capaz de arrojar nuestro sistema de modelado y simulación utilizaremos una página de datos donde el tamaño del bit será de 2x2 píxeles, tanto para la imagen codificada como sin codificar.

Compararemos los errores contados bit a bit obtenidos con cada una de las fuentes de error, contar los errores bit a bit nos ayudará a obtener resultados comparables entre los dos tipos de imagen utilizados. El análisis por símbolo y la capacidad de recuperación analizando la distancia de Hamming quedará para futuros trabajos.

En la figura 5 se muestra la evolución de los errores en las páginas codificadas y sin codificar en función del valor medio de los unos. El valor medio de los ceros se deja fijo a un valor de 300. En la gráfica se puede observar como la evolución es muy similar en ambos casos, llegando a valores de cero errores en los mismos valores de contraste, ya que el valor medio de los unos, si hemos dejado fijo el valor medio de los ceros, nos dará una idea de la separación entre ambos valores, que está relacionado con el contraste percibido.

En este caso vemos que en realidad el número de errores a nivel de bit es el mismo, o muy similar, al de los errores es muy similar. Si nos fijamos en los ejes de ordenadas tenemos la mitad de errores, aunque también, como se ha discutido anteriormente, tenemos una densidad de datos que es la mitad.

Una vez analizado el efecto del contraste, seleccionamos como punto de partida una configuración con contraste suficiente para que el número de errores sea cero en ambas imágenes. Seleccionamos un valor medio de los unos de 1100, y desde este punto vamos a añadir iluminación no uniforme con amplitud creciente y observaremos el cambio en el número de errores.

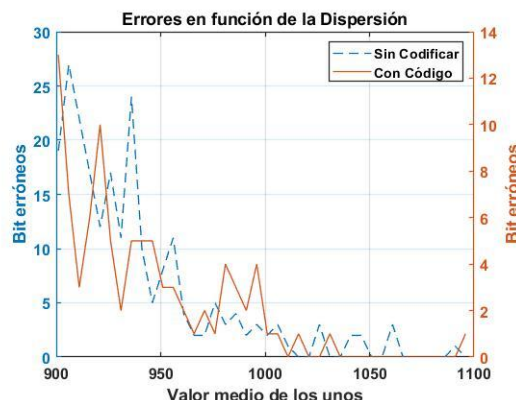


Fig.5: Errores obtenidos en función del valor medio de los unos.

En la figura 6 tenemos la evolución del número de errores en función de la amplitud de la iluminación no uniforme, donde podemos ver cómo el número de bit erróneos es sólo ligeramente superior al de la imagen sin código.

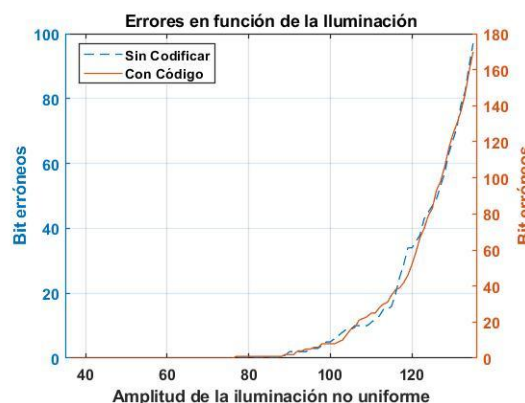


Fig.6: Errores obtenidos en función de la amplitud de la iluminación no uniforme.

Por último, volvemos a partir de una la imagen anterior en la que contamos cero errores (la imagen con contraste suficiente) y probamos a variar el radio del FPB. En la figura 7 representamos estos mismos errores en función del radio del FPB, el comportamiento tiene una evolución más abrupta en la imagen codificada, pero vemos como los errores se reducen a cero con un radio ligeramente menor del FPB.

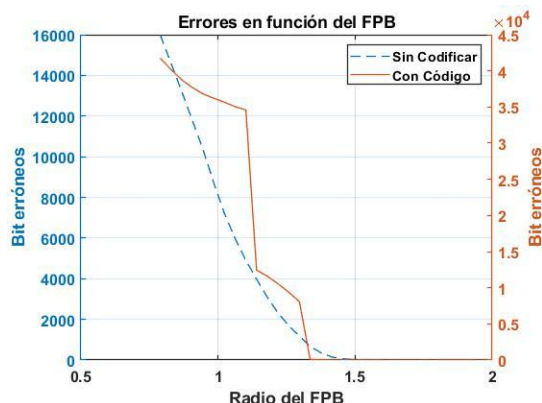


Fig.7: Errores obtenidos en función del radio del Filtro Paso Bajo ideal, donde 1 representa la Frecuencia fundamental.

## 5.- Discusión y Conclusiones

En este trabajo hemos presentado un sistema para modelar y analizar la evolución del número de errores introducidos en función de distintas fuentes de deterioro y se ha comparado el caso de utilizar una imagen con codificación o una sin codificación. Se ha encontrado que hay sutiles diferencias tanto en la evolución del número de errores como en el número total de estos, por ejemplo se ha visto que el uso de una codificación hace que la imagen sea más sensible al FPB que produce el sistema, pero si el FPB tiene apertura suficiente el número de errores mínimo se alcanza con una apertura ligeramente menor en un imagen codificada.

En futuros trabajos se analizará la posibilidad de corregir los errores actuales en los símbolos introducidos, para ver si mediante la distancia de Hamming se pueden corregir errores con lo que mejoraría la relación símbolo ruido. Incluso se puede pensar en este sistema como una forma de generar imágenes deterioradas de forma artificial para poder entrenar un supuesto sistema de filtrado basado en inteligencia artificial.

Agradecimientos: Este trabajo ha sido financiado por el Ministerio de Ciencia e Innovación con los proyectos FIS2017-82919-R (MINECO/AEI/FEDER, UE) y PID2019-106601RB-I00, la Generalitat Valenciana con los proyectos CDEIGENT/2018/024 y GV/2019/021, y la Universidad de Alicante con el proyecto UATALENTO18-10.

## Referencias

- [1] P. J. VAN HEERDEN, "Theory of optical information storage in solids", *Appl. Opt.*, 2(4), 393-400, 1963.
- [2] D. SARID, B.H. SCHECHTMAN, "A roadmap for data storage applications", *Opt. Photonics News*, 18(5), 32-37, 2007.
- [3] H. J. COUFAL, D. PSALTIS, B. T. SINCERBOX, eds. "Holographic Data Storage", (Springer-Verlag, 2000).
- [4] K. CURTIS, L. DHAR, A. HILL, W. WILSON, M. AYRES, "Holographic Data Storage: From theory to Practical Systems", (John Wiley & Sons, Ltd., 2010).
- [5] G. T. SINCERBOX, "History and physical principles", in *Holographic Data Storage*, (Springer-Verlag, pp. 3-20, 2000).
- [6] K.-I. SHIMADA, T. IDE, T. SHIMANO, K. ANDERSON, K. CURTIS, "New optical architecture for holographic data storage system compatible with Blu-ray Disc™ system" *Opt. Eng.* 53(2), 025102, 2014.
- [7] F. J. MARTÍNEZ-GUARDIOLA, A. MÁRQUEZ, E. M. CALZADO, R. FERNÁNDEZ, C. NEIPP, M. ORTUÑO, I. PASCUAL, S. GALLEGRO, "Anamorphic characterization of a PA-LCoS based holographic data storage system.", *Proc. SPIE* 10751, 1075112, 2018.
- [8] J.T. SHERIDAN et al. "Roadmap on holography", *J. Opt.* 22 (123002), 2020.
- [9] F. J. MARTÍNEZ, R. FERNÁNDEZ, A. MÁRQUEZ, S. GALLEGRO, M.L. ÁLVAREZ, I. PASCUAL, A. BELÉNDEZ, "Exploring binary and ternary modulations on a PA-LCoS device for holographic data storage in a PVA/AA photopolymer.", *Opt. Express*, 23(16), 20459-20479, 2015.
- [10] HUA-RONG GU, LIANG-CAI CAO, QING-SHENG HE, GUO-FAN JIN, "A two-dimensional constant-weight sparse modulation code for volume holographic data storage" *J. Zhejiang Univ. - Sci.C*, 12, 430-435, 2011.
- [11] MIGUEL DE CERVANTES, "Don Quijote" Descargado de <http://www.gutenberg.org/ebooks/2000> (accedido 24 de abril de 2021)
- [12] OTSU N., "A threshold Selection Method from Gray-Level Histograms." *IEEE Transactions on Systems, Man and Cybernetics*. 9 (1), 62-62, 1979.