

# **INTEGRACIÓN, DETECCIÓN Y CODIFICACIÓN DE CÓDIGOS QR DE COLORES DISTINTOS AL NEGRO**

## *INTEGRATION, DETECTION AND CODIFICATION OF QR CODES OF COLORS OTHER THAN BLACK*

***Paola Noemi San Agustín Crescencio***

Instituto Politécnico Nacional, México  
*noemi76san@gmail.com*

***Leobardo Hernández González***

Instituto Politécnico Nacional, México  
*lhernandezg@ipn.mx*

***Pedro Guevara López***

Instituto Politécnico Nacional, México  
*pguevara@ipn.mx*

***Jazmín Ramírez Hernández***

Instituto Politécnico Nacional, México  
*jramirezhe@ipn.mx*

**Recepción:** 3/noviembre/2021

**Aceptación:** 21/diciembre/2021

### **Resumen**

Los códigos QR se caracterizan por su rápida decodificación y gran capacidad de almacenamiento, sin embargo, si se desea almacenar información delicada, se debe de incorporar un sistema de seguridad. En la actualidad, la incorporación de sistemas de seguridad digitales es importante para salvaguardar la autenticidad de obras impresas. Este trabajo integra dos módulos de seguridad, además de aumentar el doble de capacidad de un solo QR. Para lograr una mayor capacidad, se encripto la información usando códigos QR de colores distintos al negro y se integraron en uno solo tipo cromático. En la fase de detección y a través de un programa desarrollado para este trabajo, se identifican los códigos QR por su color, para luego decodificar la información y recuperarla. El sistema final logra integrar e identificar códigos QR de distintos colores sin perder la información, la metodología propuesta permitirá a futuro la integración de más colores.

**Palabras Clave:** Códigos QR, integración, recuperación de datos, seguridad.

## **Abstract**

*QR codes are characterized by their fast decoding and large storage capacity, however, if you want to store sensitive information, you must incorporate a security system. Currently, the incorporation of digital security systems is important to safeguard the authenticity of printed works. This work integrates two security modules, in addition to doubling the capacity of a single QR. To achieve greater capacity, the information was encrypted using QR codes of colors other than black and integrated into a single chromatic type. In the detection phase and through a program developed for this work, the QR codes are identified by their color, and then decode the information and retrieve it. The final system manages to integrate and identify QR codes of different colors without losing the information, the proposed methodology will allow the integration of more colors in the future.*

**Keywords:** *Data Retrieval, integration, QR Codes, security.*

## **1. Introducción**

Un código QR (“Quick Response code” - código de respuesta rápida), creado inicialmente en Japón, es un código bidimensional que puede almacenar tanto datos como comentarios, servir como medio de conexión a una red inalámbrica o como enlace a diferentes URL. De acuerdo con [Ordóñez, 2012], consiste en un conjunto de puntos negros (u oscuros) ubicados según una determinada codificación, en un patrón cuadrado sobre fondo blanco (o claro). Sus características principales son: alta capacidad de codificación de datos, hasta 7.089 caracteres numéricos o 2.953 bytes. En la actualidad algunas marcas o empresas colocan sus logos en los QR, lo cual se logra eliminando porcentaje de módulos de información, que son sustituidos con la imagen o logo deseado [Villarrea, 2013].

Hay 4 niveles de corrección de errores para los códigos QR, y permiten que funcionen a pesar de la pérdida parcial de información. Cuando parte de la imagen se encuentra oscurecida, desfigurada o eliminada, estos son: Nivel L: se recupera hasta un 7% de daño, Nivel M: se recupera hasta un 15% de daño, Nivel Q: recupera hasta un 25% de daño y Nivel H: recupera hasta un 30% de daño [Qrstuff, 2011]. En aplicaciones cotidianas los códigos QR son ampliamente utilizados para el

descubrimiento de servicios básicos, además de que pueden transmitir más información durante un escaneo para aplicaciones avanzadas [Werner, 2012].

En [Langlotz, 2007], se codificó el conjunto de datos, para obtener subconjuntos denominados códigos de barras RGB 2D Datamatrix multiplexados. El método de decodificación es lento y complejo.

Es posible identificar códigos QR por medio de un color y de estos multiplexar los colores [Pandya, 2014]. De acuerdo con [Toh, 2016] tres códigos QR (QR rojo, QR verde y QR azul) se multiplexan para representar una imagen. En la figura 1 se muestra cuando se superponen entre sí, producen colores secundarios, así una imagen está formada por una mezcla de todos los colores primarios y secundarios, por el motivo de no ser estático este método, las imágenes se obtienen de un video, volviendo a este método un poco complejo en su realización física.

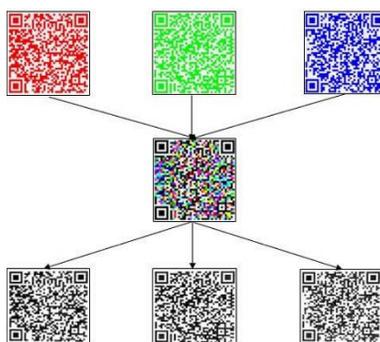


Figura 1 Multiplexación y demultiplexación de códigos QR.

En [Ferreira, 2014], También exploran la multiplexación de códigos QR, el inconveniente que se detecta es la calidad de adquisición del código impreso y la cantidad de colores generados, lo que complica la decodificación. En [Wang, 2018] se presenta una propuesta que consiste en un código QR a blanco y negro, visible a simple vista y un código oculto que solo es visible con luz infrarroja. En la figura 2 se logra apreciar el código QR a blanco y negro y el infrarrojo [Wang, 2018]. En esta propuesta se integra un nivel de seguridad que es fácil de identificar. En la propuesta anterior y en caso de copiado ilegal, solo es posible conocer un 50% de toda la información. Sin embargo, con una luz infrarroja es posible romper la seguridad de la obra original, por lo que se considera una seguridad mínima.

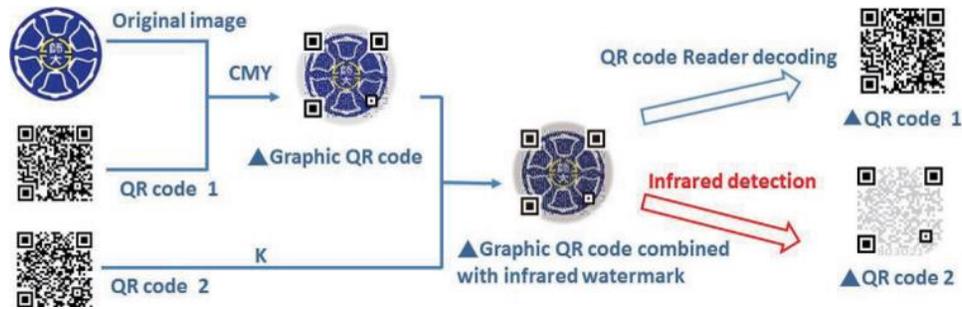


Figura 2 Producción del código QR gráfico con marca de agua QR infrarroja.

Otro método para brindar seguridad a un QR se presenta en [Godínez, 2017], se utilizan códigos QR cifrados como marcas de agua en patrones de difracción, estos se generan mediante un modelo matemático, usando la transformada de Fourier. En la figura 3 se muestran los patrones obtenidos [Godínez, 2017]. Este método brinda buena seguridad al QR, pero solo cuenta con un QR, es decir que la cantidad de información proporcionada es limitada.

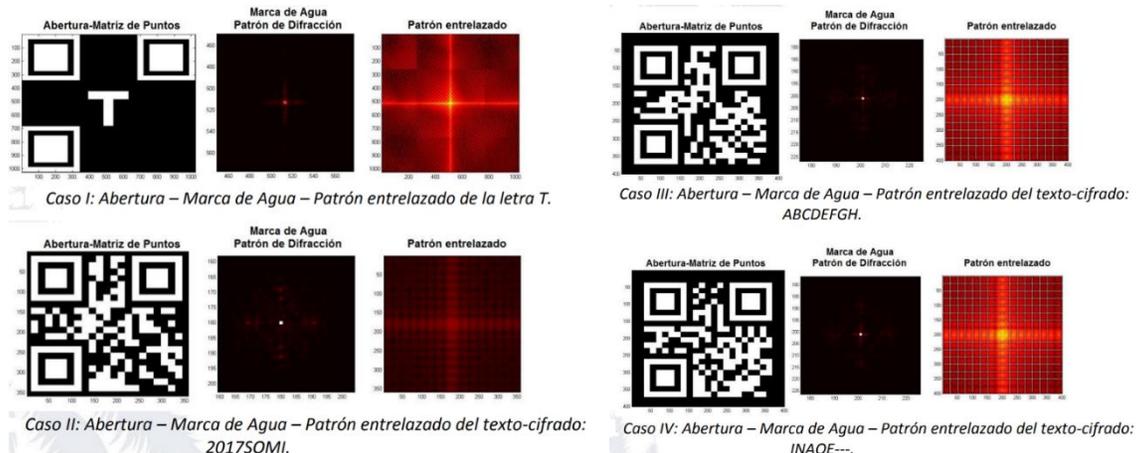


Figura 3 Patrones de difracción.

En [Rubio, 2020] se menciona la inclusión del código QR como identificación clínica asistencial (en un caso de Emergencia), debido a que generalmente las personas al momento de ingresar a un centro de atención hospitalaria lo hacen en estado de shock y con posible pérdida del conocimiento. En este tipo de aplicaciones, la integración de un método o nivel de seguridad es esencial para salvaguardar al usuario, dado que existen grupos de la sociedad que hacen mal uso de la

información, como es el vender la información personal o información de la empresa, logrando así pérdidas financieras e incluso humanas.

En [San Agustín, 2020] se menciona el diseño de un sistema de ocultamiento y recuperación de información para obras impresas, basado en la inserción de códigos QR detectables bajo luz con diferentes longitudes de onda, para otorgarle una mejor capacidad de seguridad a la información oculta en el QR. Para ello, se necesita la realización de un software con capacidad de identificar la información de dos QR a diferentes longitudes de onda contenida en un solo QR.

En este trabajo la propuesta que se presenta, integra y detecta códigos QR de colores diferentes al negro, como una estrategia de ocultamiento y recuperación de información para obras impresas, la propuesta se basa en integrar dos códigos QR en diferentes colores (rojo y azul) para la obtención de un código QR cromático, se muestra el proceso de recuperación de la información de los dos códigos QR originales, por medio de la detección de cada código QR asociado a su color.

## **2. Métodos**

El sistema de unificación y detección de código QR a partir de su longitud de onda, puede representarse a través del diagrama de estados. El diagrama de estados puede considerarse como un autómata finito determinístico, definido por un alfabeto, descrito por un conjunto de símbolos que describen gramática con contexto; esto es:  $\Sigma = \{\emptyset, A, B, C, D, E\}$ , donde:

$\emptyset$ : Símbolo nulo,

*A*: Ingreso de la información a codificar,

*B*: Generación de los códigos QR, de colores distintos al negro,

*C*: Unificación de los dos códigos QR,

*D*: Identificación de cada código QR, en QR unificado,

*E*: Decodificación y obtención de información de cada QR y fin de los estados.

En la figura 4 se muestra la estructura típica de un autómata, se compone por el alfabeto, el cual ayuda a comprender las etapas en las que se compone el sistema de unificación y detección de código QR, a partir de su longitud de onda.

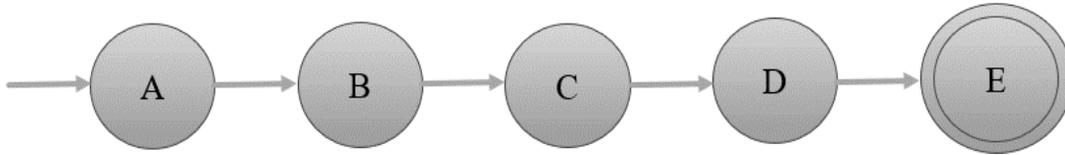


Figura 4 Diagrama de estados para representación del sistema de unificación QR.

Considerando el diagrama de estados de la figura 4, se observa que es posible formar las siguientes palabras de la gramática con contexto: *ABCDE* y *ABC*, éstas son solo algunas, las cuales permiten entender la evolución de un estado a otro. Como se puede observar la relación de los estados es lineal, se comienza en *A* donde se determina la información a codificar. Al pasar al estado *B* se generan los dos códigos QR de colores distintos al negro, en este caso rojo y azul. En el estado *C* se unifican los dos códigos QR generados en el estado *B*.

Los siguientes dos estados son parte de la recuperación de la información; en *D* se identifica cada QR (QR azul y QR rojo) que están en el QR unificado. En *E* se decodifica cada QR identificado y se obtiene la información. Para el desarrollo del software se utilizaron bibliotecas de Python, como se observa en la tabla 1. Otros datos necesarios para este proyecto se ven en la tabla 2.

En la figura 5 se presenta el diagrama de flujo para la generación del código. La propuesta se divide en tres etapas principales, las cuales deben de cumplir los siguientes objetivos.

Tabla 1 Bibliotecas utilizadas de Python.

Biblioteca	Uso
<i>segno</i>	Generación de QR de color
<i>cv2</i>	Se utiliza para leer imágenes y procesarlas en tiempo real, en este caso leer la imagen creada de los QR a color
<i>numpy</i>	Proporciona estructuras de datos, en este caso para la realización de cálculos con las matrices de las imágenes.
<i>pyzbar</i>	Leer el código QR
<i>PIL</i>	Permite la edición de imágenes, en este caso las imágenes de los códigos QR, en formato PNG.

Tabla 2 Datos relevantes del software.

Dato	Uso
<i>qr0</i>	QR rojo
<i>qr1</i>	QR azul
<i>res1</i>	Escalamiento de la imagen QR rojo PNG
<i>res2</i>	Escalamiento de la imagen QR azul PNG
<i>QR_integrado</i>	Los dos QR superpuestos en el mismo espacio
<i>rojo_bajos</i>	Todos los colores en rojo tonalidad baja
<i>rojo_altos</i>	Todos los colores en rojo tonalidad alta
<i>azul_bajos</i>	Todos los colores en azul tonalidad baja
<i>azul_altos</i>	Todos los colores en azul tonalidad alta
<i>mask1</i>	Detención de pixeles, dentro del rango de tonos bajos y altos del color rojo
<i>mask2</i>	Detención de pixeles, dentro del rango de tonos bajos y altos del color azul

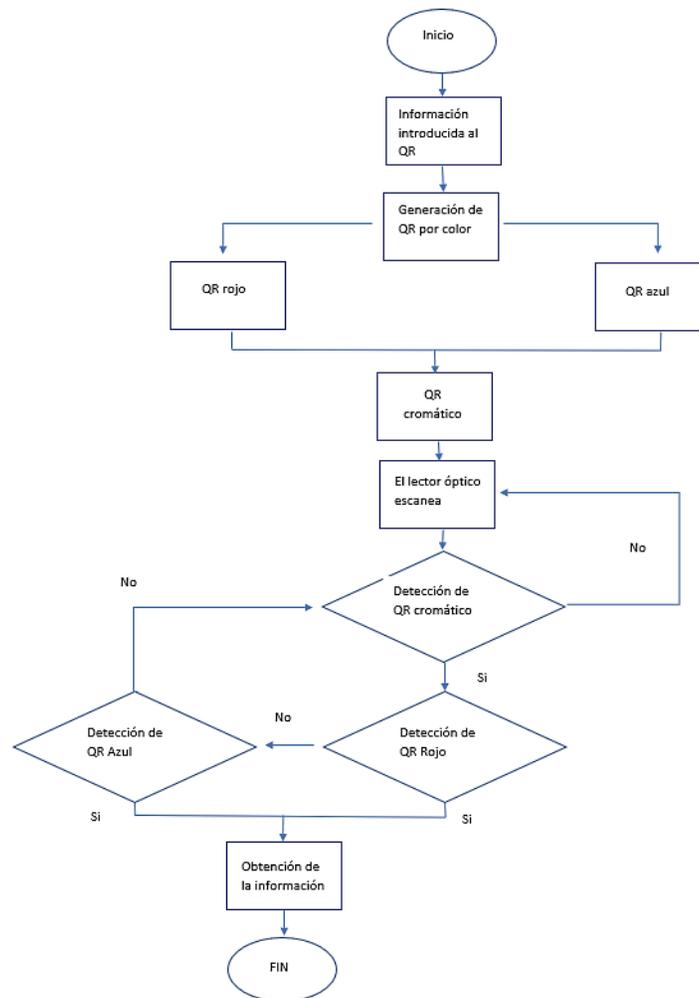


Figura 5 Diagrama de flujo del sistema.

### Primera etapa: generación e integración de los códigos QR de color

Se inicia creando dos códigos QR, uno de color rojo (qr0) y el otro de color azul (qr1), después se unifican los dos códigos QR en uno solo, cuidando que si el espacio que ocupa en la matriz un píxel azul es el mismo espacio que ocupa un píxel rojo en la matriz, al momento de crear el QR cromático (QR\_integrado), los pixeles no se deben sumar, se unifican, como se muestra en la figura 6.

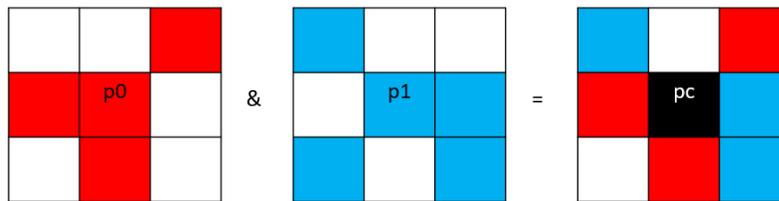


Figura 6 Unificación del píxel rojo y azul.

### Segunda etapa: detección de los códigos QR de color originales

Del QR cromático obtenido se identifica cada código QR por su color original, teniendo presente el nuevo color generado (negro). La imagen del QR cromático se convierte a modelo HSV (“Hue, Saturation, Brightness” - Matiz, Saturación, Brillo), para delimitar el color que se desea identificar, en este caso se limitan a tonalidades bajas y altas del rojo, tonalidades bajas y altas del azul, y el negro. Se crea una nueva matriz, en la cual se va activando cada píxel encontrado de acuerdo con el color que se desea. En la figura 7 se observa la identificación del QR rojo.

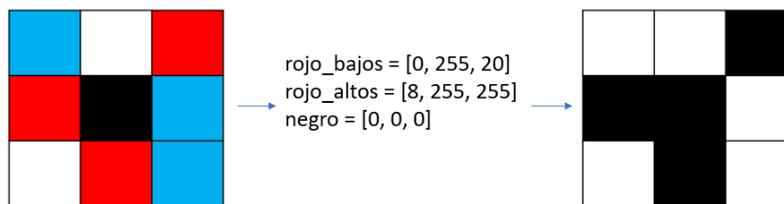


Figura 7 Identificación del código QR rojo.

### Tercera etapa: validación de la información del QR original

Por último, se verifica que la información obtenida de la nueva matriz sea la misma que la información original en el QR rojo o azul. En la figura 8 se muestra un ejemplo de validación de la información.

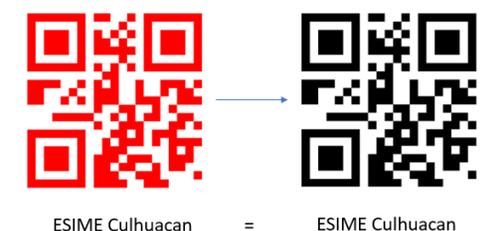


Figura 8 Validación de la información.

### 3. Resultados

Se presentan los principales resultados obtenidos de las tres etapas más importantes del sistema: Generación e integración de los códigos QR de color, detección de los códigos QR de color originales y decodificación para validación de la información del QR original, una etapa primordial para que las etapas restantes funcionen adecuadamente, es lograr la unificación de los dos códigos QR de diferente color. Como caso de estudio se consideran el enlace del canal de YouTube de la escuela ESIME Culhuacan y el enlace de la escuela ESIME Culhuacan.

#### Fase 1: generación e integración de códigos QR de colores azul y rojo

Se utiliza el lenguaje de programación Python para la programación del software. Para esta fase se utiliza la biblioteca *segno*, la cual permite la generación de códigos QR a colores. En la figura 9 se observan la generación de los códigos QR-Rojo y QR-Azul, el QR-Rojo contiene el enlace del canal de YouTube de la unidad ESIME Culhuacan y el QR-Azul contiene el enlace de la unidad ESIME Culhuacan.

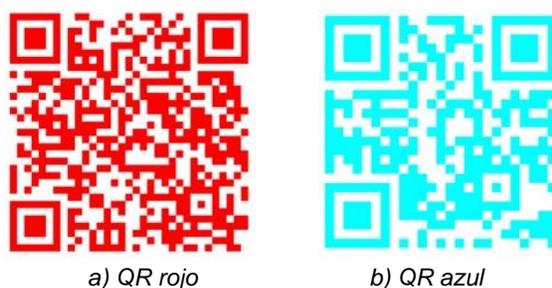


Figura 9 QR distintos al negro generados con la biblioteca *segno* de Python.

En la figura 10, Se observa que los dos QR se encuentran unificados en uno único.



Figura 10 Código QR integrado.

## Fase 2: detección de los originales QR-Azul y QR-Rojo en QR integrado

Para esta fase se realizó un programa que detecta los colores rojo, azul y negro, la razón de incorporar la detección del negro se debe a la combinación de los colores, como se muestra en la figura 10; en esta se procesa para la identificación de los colores, para esto se utilizan librerías *opencv* y *PIL*. En la tabla 3, se presentan los colores y sus combinaciones. En la tabla 4 se muestran los colores del QR integrado y que color ocuparían en la nueva matriz, en representación del QR recuperado. En la figura 11 se muestra el proceso de identificación de colores y recreación del QR que se identifica, quitando el ruido que adquirió.

Tabla 3 Colores a procesar.

Color	Combinación	QR original
rojo	No	rojo
azul	No	azul
negro	Si	Rojo y azul

Tabla 4 Colores a procesar.

Color	Representación Hexadecimal (0XRRGGBB)	Píxel negro (ocupa un lugar en la matriz)
rojo	0x00FFFF	Si
azul	0xFFFF00	Si
negro	0x000000	Si

## Fase 3: decodificación y validación de la información del QR original

Una vez identificado cada color, se procede a la etapa de decodificación de cada QR recreado como son: *qr\_RNcode* (QR-Rojo) y *qr\_ANcode* (QR-Azul), para extraer la información que se ocultó, para esto se utilizan las bibliotecas *pyzbar* y *decode*.

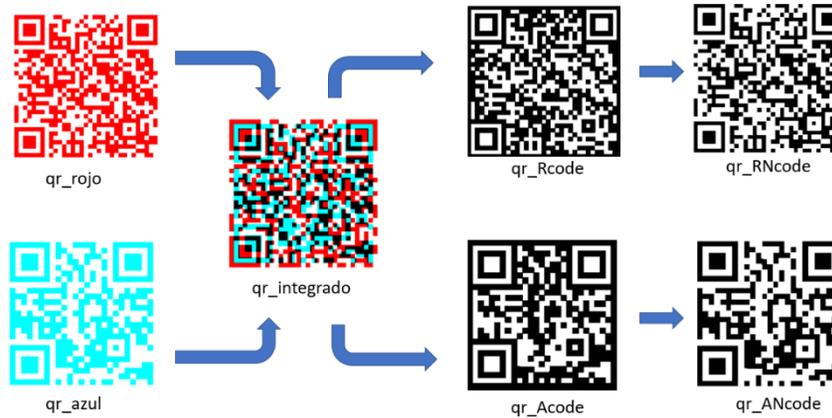


Figura 11 Proceso de identificación de códigos QR.

En la figura 12 se visualizan los dos enlaces que se encriptaron en un inicio, como se indica en las imágenes, se logró la recuperación correcta de cada información que se ocultó.

```
D:\Identificacion_QR\venv\Scripts\python.exe D:/Lector_QR/Lector_QR.py
https://www.youtube.com/channel/UckdLps4q7WbnBhrBkbvU0yA
Process finished with exit code 0
```

QR rojo: Enlace de canal de YouTube de ESIME Culhuacan

```
D:\Identificacion_QR\venv\Scripts\python.exe D:/Lector_QR/Lector_QR.py
https://www.esimecu.ipn.mx
Process finished with exit code 0
```

QR azul: Enlace de la página de ESIME Culhuacan

Figura 12 Recuperación de la información.

Los dispositivos móviles cuentan con lectores de códigos QR, ya sea como una aplicación o como función en la aplicación “cámara” del dispositivo, en la tabla 5 se muestra que código QR puede detectar el sistema operativo Android, dependiendo su versión.

Tabla 5 Detección de QR de colores con un dispositivo móvil.

Detención de QR de color en Android 6.0 al 8.1	Detención de QR de color en Android 9.0 al 12.0	Lo que detecta Android 6.0 al 8.1, en el QR cromático	Lo que detecta Android 9.0 al 12.0, en el QR cromático
 qr_rojo	 qr_rojo		 qr_rojo
	 qr_azul		
	 qr_Rcode		
	 qr_Acode		
 qr_RNcode	 qr_RNcode		
 qr_ANcode	 qr_ANcode		

## **4. Discusión**

El desarrollo de este proyecto más que una innovación es una necesidad, dado que cada día el uso de los códigos QR son más indispensables en todas las áreas, por este motivo es necesario brindarle seguridad a la información que se distribuirá. Las ventajas de este trabajo son: mayor almacenamiento de información al usar dos códigos QR, que solo uno, la metodología empleada solo otorga la información del QR de color que se desea conocer, no de los dos QR. La desventaja es que algunos dispositivos móviles que tienen la herramienta de escaneo son capaces de detectar el QR más oscuro, aunque se encuentren unificados, por lo que es necesario la integración de un sistema de seguridad extra.

## **5. Conclusiones**

Después de analizar los resultados, se determina que es posible la integración de dos códigos QR de colores distintos al negro, en este caso el rojo y azul; sin embargo, la técnica empleada permite integrar otros colores como el amarillo y verde. Al utilizar otro tipo de colores se sugiere realizar algunas pruebas, ya que el rojo, azul y amarillo son colores primarios que al combinarlos generan colores secundarios y esta combinación de color se crea al momento de integrar los códigos QR en uno solo; la importancia de tomar en cuenta esta combinación es al momento de identificar cada código QR, de lo contrario se obtendría un código QR erróneo. Esta forma de seguridad a la información es conveniente para la validación de obras impresas, dado que aporta información más completa comparado a la técnica de marca de agua, aporta más información que un solo código de encriptación como lo son el código de barras y un solo código QR. Estos resultados permiten validar esta propuesta, para una integración posterior de encriptación de la información a ocultar, por lo que se tendrá un sistema más robusto en cuanto a seguridad.

## **6. Bibliografía y Referencias**

- [1] Godínez, A. P., Meléndez, R. P., & Treviño-Palacios, C. G., (2017). Códigos QR cifrados como Marcas de Agua en Patrones de Difracción. In *Somi XXXII*, Congreso De Instrumentacion.

- [2] André, P. S., & Ferreira, R. A. S., (2014). Colour multiplexing of quick-response (QR) codes. *Electronics Letters*, 50(24), 1828-1830.
- [3] Langlotz, T., & Bimber, O., (2007). Unsynchronized 4D barcodes. In *International Symposium on Visual Computing*, pp. 363-374. Springer, Berlin, Heidelberg.
- [4] Ordóñez, J. L., (2012). Códigos QR. *Manual formativo de ACTA*, (63), 9-28.
- [5] Pandya, K. H., & Galiyawala, H. J., (2014). A Survey on QR Codes: in context of Research and Application. *International Journal of Emerging Technology and Advanced Engineering*, 4(3), 258-262.
- [6] Qrstuff.com, (2011). QR Code Error Correctio: <http://www.qrstuff.com/blog/2011/12/14/qr-code-error-correction>. [Accessed: 15- Sep- 2021].
- [7] Rubio Juárez, H. U. G. O. (2020). La Inclusión Del Código Qr, En El Documento Nacional De Identidad.
- [8] San Agustín Crescencio P. N. y Osio Arrazola J., (2020). Diseño de Ocultamiento y Recuperación de Información en Obras Impresas Utilizando Códigos QR, Luz de Diferentes Longitudes de Onda y una Computadora de Placa Reducida.
- [9] Toh, S. R., Goh, W., & Yeo, C. K., (2016). Data exchange via multiplexed color QR codes on mobile devices. In *2016 Wireless Telecommunications Symposium (WTS)*, pp. 1-6. IEEE.
- [10] Villarrea, O., & Villamizar, R., (2013). Incrustación de imágenes en códigos de barras bidimensionales de rápida respuesta QR-codes. *Revista vínculos*, 10(2), 277-288.
- [11] Wang, Y. M., Sun, C. T., Kuan, P. C., Lu, C. S., & Wang, H. C., (2018). Secured graphic QR code with infrared watermark. *IEEE International Conference on Applied System Invention (ICASI)*, IEEE, pp. 690–693.
- [12] Werner, M., & Schönfeld, M., (2012). DualCodes: Backward Compatible Multi-layer 2D-Barcodes. In *International Conference on Mobile and Ubiquitous Systems: Computing, Networking, and Services*, pp. 25-36. Springer, Berlin, Heidelberg.