



Instituto Tecnológico de Culiacán

Ing. Electrónica

Tarea 1: Investigación sobre patrón de Bayer y Demosaicing (o Reconstrucción)

Moisés Ezequiel Domínguez Salcedo 17170527

Materia: Procesamiento digital de señales

Grupo: 8:00 a 9:00 hrs.

Profesor: Dr. Modesto Guadalupe Medina Melendrez

Filtros y algoritmos de reconstrucción .....	3
Filtro de Bayer .....	3
Demosaicing .....	5
Algoritmos de demosaicing .....	5
No-Adaptativos .....	5
Réplicas de Vecino Cercano .....	5
Interpolación Bilineal .....	6
Transición de Tonalidad suavizada .....	6
Transición Logarítmica de Tonalidad suavizada .....	6
Algoritmos Adaptativos .....	7
Interpolación con Detección de Bordos .....	7
Interpolación con Corrección de Color .....	8
Interpolación basada en Umbral Variable según un Número Fijo de Gradientes .....	8
Patrones Conocidos .....	9
Patrón de Borde Superior: .....	9
Patrón de Borde Inferior: .....	9
Patrón de Raya .....	9
Patrón de Esquina: .....	9
Explicación de algoritmo no adaptativo, Bilinear Demosaicing ( high-quality linear interpolation for demosaicing of bayer-patterned color images) .....	10
Otros tipos de patrón de Bayer o alternativas al sensor .....	11
Patrón de Bayer con celdas “pancromáticas” .....	11
Filtro Foveon X3 .....	11
Filtro CYGM .....	12
Conclusión .....	12
Referencias .....	12

## Filtros y algoritmos de reconstrucción.

Los siguientes conceptos fueron extraídos, de una recopilación de la siguientes ligas: [https://es.wikipedia.org/wiki/Mosaico\\_de\\_Bayer](https://es.wikipedia.org/wiki/Mosaico_de_Bayer), <https://sci-hub.se/https://doi.org/10.1117/12.561568>, algunas otras fueron reseñas de distintos autores, en el caso de los algoritmos de demosaicing o reconstrucción, los extraje el autor Henrique S, Malvar, Li-Wei He, and Ross Cutler titulado high-quality linear interpolation for demosaicing of bayer-patterned color images

### Filtro de Bayer

El filtro, máscara o mosaico de Bayer es un tipo de matriz de filtros, rojos

R	G	R	G
G	B	G	B
R	G	R	G
G	B	G	B

Figure 1: Typical Bayer filter pattern showing the alternate sampling of red, green and blue pixels.

verdes y azules, que se sitúa sobre un sensor digital de imagen (CCD o APS) para hacer llegar a cada fotodiodo la información de luminosidad correspondiente a una sección de los distintos colores primarios. Interpolando las muestras de cuatro fotodiodos vecinos se obtiene un píxel de color.

- Se llama así por su creador, Bryce Bayer, de la empresa Eastman Kodak.
- En su composición está formado normalmente por un 50 % de color verde , 25 % rojo y verde.
- También llamados CFA (Color filter Array ) o patrón de mosaicos.
- Usados para maximizar la capacidad de percibir los filos o rebordes mas finos en los canales de iluminación, necesitando ser el fitro en su mayoría de color verde (50%)
- Los filtros Bayer provocaron gran cambio en como funcionaban las cámaras digitales obteniendo métodos diferentes para obtener imágenes RGB mediante la interpolación

La siguiente imagen corresponde a la patente de un filtro Bayer de un APS (active pixel sensor), donde se muestra la construcción desde el semiconductor de un filtro con tipo bayer Bayer. Merrill, R. B. (1999)

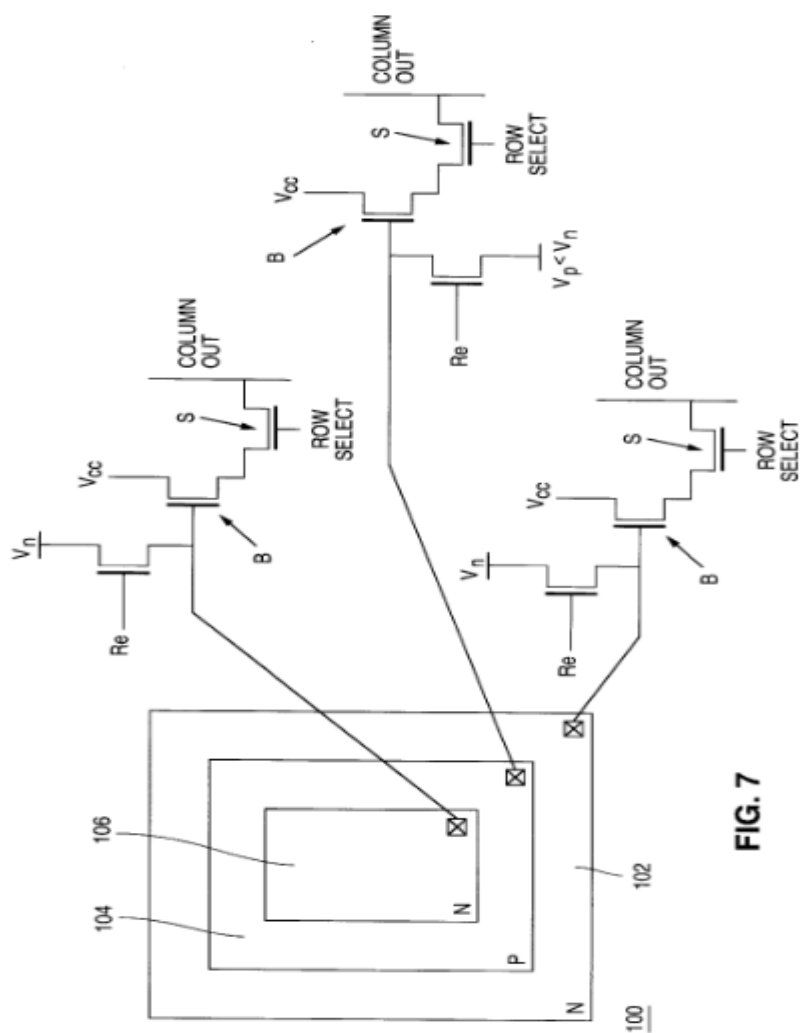
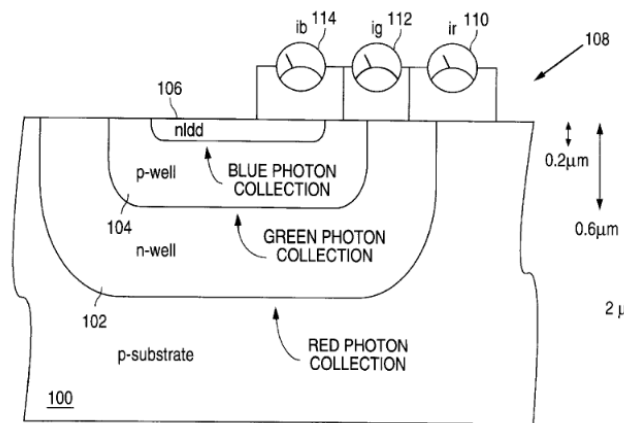


FIG. 7

## Demosaicing

Después de la captación de la imagen y una vez filtrada por el filtro de Bayer, donde tenemos una única componente de color en cada píxel, se aplica un proceso de demosaicing o reconstrucción de la imagen, con el fin de tener los tres componentes de color en cada píxel para tener así la imagen en color RGB. Para llevar esto a cabo, se aplica a la imagen Bayer, uno u otro algoritmo, según la calidad que queramos obtener y la potencia de nuestro procesador.

## Algoritmos de demosaicing

Como ya hemos dicho, para reconstruir la imagen a partir de la Imagen Bayer, debemos efectuar un proceso de demosaicing a través de algoritmos más o menos complejos que nos darán resultados más o menos precisos. Los algoritmos simples nos darán un resultado pobre y de baja calidad, pero el coste procesal y por lo tanto, la energía empleada en el cálculo, será muy pequeña; por el contrario, de los algoritmos más complejos obtendremos una calidad más óptima a cambio de un coste procesal más elevado y por lo tanto, un coste energético, y la necesidad de un procesador más potente.

Básicamente hay dos tipos de algoritmos de demosaicing, los No-Adaptativos y los Adaptativos:

### No-Adaptativos

Los Algoritmos No-Adaptativos son aquellos que aplican una algorítmica fija independientemente de la imagen o parte de la imagen que procesen. Son los algoritmos más sencillos, ya que no hacen ningún estudio previo de las características de la imagen y por lo tanto, siempre aplican el mismo procesamiento. Así pues, gracias a su sencillez, son muy fácilmente implementables y tienen un coste procesal y energético bajo. Por el contrario, al no adaptarse a las características de la imagen, tienen una calidad baja.

### Réplicas de Vecino Cercano

Este es el procesamiento más sencillo y simplemente se basa en recorrer la imagen, y en los píxeles donde no hay información de un componente RGB, se toma el mismo valor del píxel más cercano del mismo componente y se copia. Si

hay dos o más píxeles del componente en cuestión a igual distancia, se sigue una regla aleatoria de selección (por ejemplo, siempre priorizamos los píxeles de una fila superior y una columna más a la izquierda). Evidentemente, es el algoritmo más rápido, ya que no se hace ni un solo cálculo, simplemente llenan los huecos vacíos.

### Interpolación Bilineal

Este procesamiento es muy similar al anterior, pero en vez de copiar el mismo valor del píxel más cercano del mismo componente de color, se miran los valores de los píxeles más cercanos, y como normalmente habrá 2 o 4 píxeles a la misma distancia, se hace la media aritmética y el resultado es el nuevo valor del píxel.

### Transición de Tonalidad suavizada

Lo que pretende este algoritmo es corregir uno de los problemas de la Interpolación Bilineal, y es que, creaba relieves de color en la interpolación de los componentes azul y rojo de los píxeles. En las imágenes, los píxeles verdes básicamente llevan la información de luminosidad, y los rojos y azules del color o la tonalidad. Lo que hace el algoritmo de Transición de Tonalidad suavizada es tratar la imagen para canales de color. Así, el canal verde aplica una interpolación lineal que se debe hacer al principio, porque del resultado de esta depende el de los otros canales. En el canal azul y en el rojo, lo que hace es calcular la media de los píxeles del componente, pero dividido por el valor del píxel verde en aquella posición. Así se suaviza según la cantidad de luz (canal verde) el nivel de tonalidad o color (canales rojo y azul) y se reduce la creación de falsos relieves de color.

### Transición Logarítmica de Tonalidad suavizada

Se hace lo mismo que en el algoritmo anterior, pero se trabaja con el valor logarítmico de los píxeles (si un píxel tiene valor de verde  $G$ , se transforma en valor de verde  $G' = \log(G)$ ). Primero se calcula el canal verde con los valores en logaritmos y luego se calculan los canales azul y rojo como la media de los píxeles del canal dividido por el valor de verde en ese píxel (todo en logaritmos). Finalmente, con los valores finales de los tres componentes  $R'G'B'$  de cada píxel, se les hace el antilogaritmo, para recuperar el valor correcto expresado de forma

lineal RGB. El hecho de trabajar en base logarítmica hace que se suavice aún más la creación de falsos relieves de color.

### Algoritmos Adaptativos

Los Algoritmos Adaptativos son, por el contrario, algoritmos más complejos. Estos hacen un análisis de las características de la imagen o parte de la imagen y, dependiendo de estos parámetros evaluados, aplican un tipo de procesamiento u otro para que se adecue a esta imagen y poder así obtener el resultado óptimo. Estos algoritmos son más complejos y por lo tanto tendrán un coste procesal y energético mayor. No obstante, es evidente que se obtendrá un resultado mucho más exacto y de más alta calidad.

### Interpolación con Detección de Bordes

El Sistema Visual Humano es muy sensible a los bordes y a los relieves que forman las imágenes, y es justamente en los bordes donde los Algoritmos No-Adaptativos tienen las carencias más notables. Así que, el Algoritmo Adaptativo, primero fue aquel que detectaba bordes para aplicar en esa zona el procesamiento más adecuado. De hecho, en la Interpolación con Detección de Bordes, no se crea ningún algoritmo nuevo, sino que primero se hace un "barrido" de la imagen y se aplica un Algoritmo de Detección de Bordes y, según si hay o no un relieve, se da un valor a una variable umbral por aquel píxel. Después se hace el procesamiento del demosaicing, primero por el canal verde, donde, en cada píxel, según el valor del umbral que indicará si pertenece a una zona homogénea, una zona donde hay un borde vertical o una zona donde hay un borde horizontal, se aplica el procesamiento no-adaptativo más conveniente. Así, si estamos en una zona homogénea, se aplica el algoritmo de Interpolación Bilineal tanto en los píxeles verde vecinos de arriba, de abajo, de la derecha y de la izquierda. Si, en cambio, estamos en un borde vertical, se aplica el algoritmo de interpolación bilineal sólo con los píxeles verdes de la izquierda y la derecha (ya que es en horizontal donde está la homogeneidad). Y si, finalmente, tenemos un borde horizontal, se calcula con los píxeles verdes en sentido vertical. Para los canales azul y rojo se pueden adoptar varias variantes, pero siempre se aplica un algoritmo no-adaptativo, ya sea el de Transición de Tonalidad suavizada (logarítmica o no) o simplemente el de

Interpolación Bilineal. Donde sí que podemos ajustar el procesado, es en la mejora del algoritmo de Detección de Bordes, ya que es la base de este proceso de demosaicing.

### Interpolación con Corrección de Color

Este algoritmo y los que siguen son ya de un grado de complejidad más elevada, que implicará un coste procesal mayor. Siguen aplicando algoritmos de Detección de Bordes, que hacen que el algoritmo final de demosaicing se decante por un código u otro respecto una variable umbral que crea el algoritmo de detección de bordes. La gran diferencia es que, en el cálculo del canal verde, no sólo utilizamos los píxeles verdes en dirección horizontal, vertical o todos según si tenemos un relieve vertical, horizontal o estamos en zona homogénea respectivamente, sino que incluimos en el cálculo del píxel verde (de una posición donde en la Imagen Bayer había un píxel azul) el valor de un gradiente (Derivada Laplaciana de Segundo Orden) de los píxeles azules vecinos. En los píxeles donde en la Imagen Bayer había rojo, se incluye un gradiente rojo en vez de un gradiente azul. Para calcular el valor de azul o rojo en un píxel donde en la Imagen Bayer había verde, también se crean unos gradientes que dependen de los valores de verde de los píxeles vecinos (los que hemos calculado primero) y los valores de los píxeles vecinos del componente azul o rojo.

Así que, en definitiva, con este algoritmo, no sólo se tiene en cuenta el hecho de si hay o no un borde, sino que además se añade la interrelación de los valores de los píxeles en las tres capas de color.

### Interpolación basada en Umbral Variable según un Número Fijo de Gradientes

Este algoritmo complejo se basa en crear un Umbral que variará su valor dependiendo de qué gradiente hayamos utilizado para su cálculo. El Gradiente (hay ocho gradientes posibles que corresponden a las direcciones o puntos cardinales; norte, sur, este, oeste, noreste, sureste, noroeste, suroeste), se escoge según la dirección dentro de una subimagen (matriz de 5x5 píxeles). Cada conjunto de 8 gradientes determina un umbral y este umbral determina otro subconjunto de gradientes (norte, sur, este, oeste). Los gradientes del primer conjunto, que tienen un valor bajo, indican que se trata de una zona con color de



píxel similar. Los gradientes del subconjunto que, por el contrario, tienen valores altos, indican que es una región donde hay muchos detalles o relieves pronunciados. Los gradientes del subconjunto se utilizan para localizar zonas con características similares. El valor general de los píxeles de estas zonas similares determinará el valor particular de los píxeles de la zona.

### Patrones Conocidos

En este algoritmo, nos volvemos a fijar una vez más en la información de luminancia, es decir, en el canal de píxeles verdes. Lo que tenemos aquí son cuatro posibles patrones conocidos en una subimagen de 3x3 píxeles, donde el píxel central correspondería a una celda en la cual, en la imagen Bayer, habría valor de azul o rojo, por lo que, encima, debajo, izquierda y derecha (forma de cruz), tendríamos píxeles de color verde. Los patrones posibles serían:

**Patrón de Borde Superior:** El valor de los píxeles de arriba, de la izquierda y de la derecha es mayor que el valor medio de los cuatro, y el valor de abajo es menor que la media de los cuatro.

**Patrón de Borde Inferior:** El valor de los píxeles de abajo, de la izquierda y de la derecha es mayor que el valor medio de los cuatro, y el valor de arriba es menor que la media de los cuatro.

**Patrón de Raya:** El valor de los píxeles de la izquierda y de la derecha es mayor que el valor medio de los cuatro, y el valor de arriba y de abajo es menor que el valor medio de los cuatro.

**Patrón de Esquina:** El valor de arriba y de la izquierda es mayor que el valor medio de los cuatro, y el valor de abajo y de la derecha es menor que la media de los cuatro.

Lo que hace es recorrer el canal verde de la imagen, y donde falta un píxel verde (si los píxeles vecinos tienen la estructura de los patrones de borde), se aplica un algoritmo dentro de la subimagen 3x3 según como sea esta subimagen. Pero si corresponde a un patrón de raya, se utiliza otro algoritmo que analiza una subimagen mayor de 5x5, para verificar que sea realmente una línea continua. Si

por último tiene la forma del patrón de la esquina, también coge una subimagen de 5x5, pero solo analiza los píxeles que siguen la teórica diagonal de la esquina. Si la imagen, por el contrario, no corresponde a ningún patrón y es homogénea, se aplica una interpolación bilineal. Por los canales azul y rojo, se pueden aplicar algoritmos no-adaptativos, como el de interpolación bilineal o el de transición de tonalidad suavizada.

#### Explicacion de algoritmo no adaptativo, Bilinear Demosaicing ( high-quality linear interpolation for demosaicing of bayer-patterned color images)

En las reconstrucción bilineal el valor verde  $g(i,j)$  en la posición  $(i,j)$  termina siendo rojo o azul, y es calculado por el promedio de los colores aledaños al color verde en un patrón de cruz como el que tenemos en la figura 1, la ecuación representa de manera matemática como es que algoritmo bilineal realiza la reconstrucción.

- Primer selecciona y suma la cantidad de un color específico que se repita dentro de un pixel, partiendo de eso debe haber en total dos verdes uno rojo y uno azul, utilizando valores aledaños y calculando el promedio de cuanta vez aparece el verde podemos encontrar que valor es el que resulta en un alrededor en el caso de la figura 1, serian cuatro pixeles. cruz como lo muestra la imagen.

$$\hat{g}(i,j) = \frac{1}{4} \sum_{(m,n) \in \{(0,-1), (0,1), (-1,0), (1,0)\}} g(i+m, j+n) \quad (1)$$



**Figure 1.** Typical Bayer mosaic for color image capture in single-CCD digital cameras; the G subimage has twice as many pixels as the R and B subimages.

## Otros tipos de patrón de Bayer o alternativas al sensor

Por lo que pude ver la mayoría de los filtros utilizan las mismas técnicas de demosaicing dado que, lo que cambian son los tipos de colores empleados, pero la matriz o acomodo sigue siendo el mismo.

### Patrón de Bayer con celdas “pancromáticas”

Una manera en la que los mismos desarrolladores de Kodak se dieron cuenta de que mejoraba el filtro de bayer, en cuanto al incremento de la sensibilidad de la luz en el sensor, era incluir algunas celdas llamadas pancromáticas, que son capaces de capturar todas las componentes frecuenciales (todas las longitudes de onda) de la luz visible, es decir, celdas blancas. A este nuevo filtro se le llamó filtro RGBW (Red, Green, Blue, White). También existe el filtro CMYW que es el análogo en colores sustractivos.

### Filtro Foveon X3

Otra manera de separar el color es el sensor Foveon X3. Este tiene tres niveles de silicio diferenciados, cada uno de los cuales captan un componente de color RGB (una longitud de onda) diferente. Así resulta una matriz de color diferente en cada uno de los niveles del sensor, sin utilizar ningún filtrado, y, por tanto, sin perder luminosidad una representación de ello sería la sig. imagen:

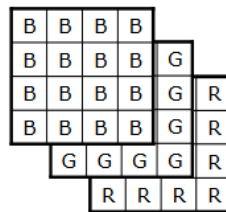


Figure 3: Schematic depiction of Foveon X3 image sensor showing stacks of color pixels, which record color depth wise in silicon.

lor planes that result directly from image capture, without interpolation.

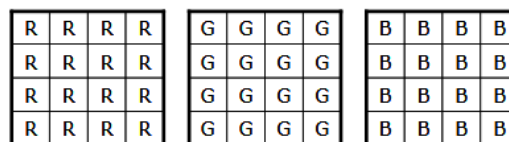


Figure 4: Fully populated image sampling found in film scanners, color-separation prism cameras, and also in Foveon X3 image sensors.

## Filtro CYGM

El sistema es parecido al Bayer, pero trabaja con cyan y amarillo en las líneas impares y utiliza verde y magenta en las líneas pares, es capaz de tener mayor rango dinámico porque la luminancia es mejor que el Bayer, pero no es tan preciso en el color.

## Conclusión

Como tal, un patrón tipo Bayer sirve para complementar una imagen RGB, con el avance del tiempo estos se han ido mejorando, al punto donde tiene mejores algoritmos de reconstrucción cada vez mas complejos obteniendo mejores imágenes, a su vez estos filtros sirven para maximizar características que en un pasado posiblemente hubiese sido impensable.

## Referencias

Hubel, P. M., Liu, J., & Guttosch, R. J. (2004, June). Spatial frequency response of color image sensors: Bayer color filters and Foveon X3. In *Sensors and Camera Systems for Scientific, Industrial, and Digital Photography Applications V* (Vol. 5301, pp. 402-407). International Society for Optics and Photonics.

Merrill, R. B. (1999). U.S. Patent No. 5,965,875. Washington, DC: U.S. Patent and Trademark Office.

Wikipedia