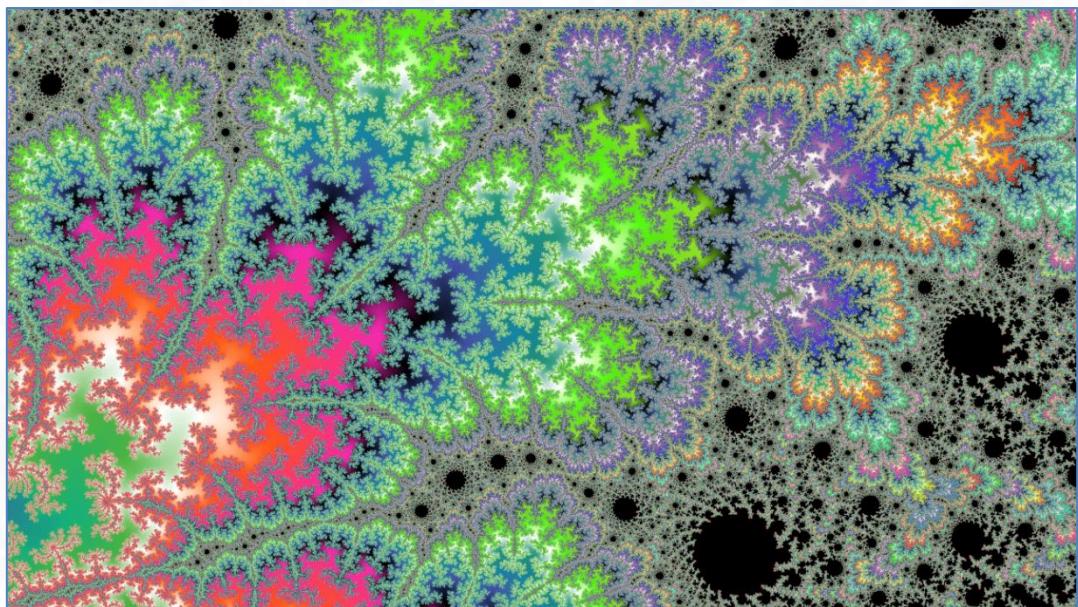


# Algoritmo de Paleta extendida

Sergio CT



« La mente que se abre a una nueva idea jamás vuelve a su tamaño original ...»

**Albert Einstein**

## Contenido

<b>Introducción</b> .....	<b>3</b>
<b>Motivación y objetivos</b> .....	<b>3</b>
<b>Algoritmo</b> .....	<b>4</b>
• Generación dinámica de la paleta extendida .....	4
• Asignación de color mediante iteración suavizada .....	5
1. Selección del par de colores de la paleta base .....	5
2. Generación dinámica del color rotado .....	6
3. Mezcla entre los colores base y extendidos .....	7
4. Interpolación final entre los dos colores mezclados .....	7
<b>Conclusiones</b> .....	<b>8</b>

## Introducción

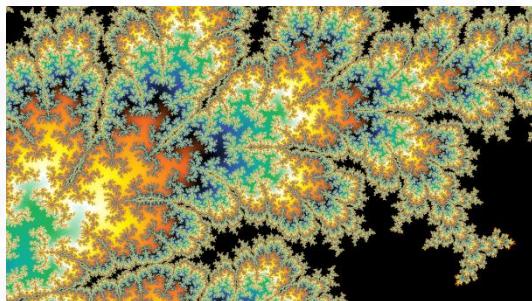
Este documento describe un método para ampliar dinámicamente una paleta de colores de forma que se elimine la repetición perceptible de patrones cromáticos típica de los fractales generados mediante algoritmos de tiempo de escape. El objetivo es mantener una coloración continua, suave y visualmente coherente durante el proceso de zoom, incluso cuando el valor de iteración crece sin límite.

Este trabajo se inspiró en una idea propuesta originalmente por **Christian Kleinhuis**, quien amablemente compartió un avance de su fórmula antes de su publicación. Durante nuestra conversación, señaló que no se conocían implementaciones capaces de abordar este problema de forma totalmente satisfactoria, lo que motivó el desarrollo de este algoritmo alternativo e independiente.

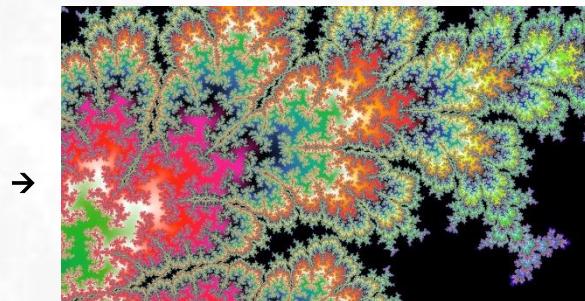
Para más información sobre la propuesta original de Christian Kleinhuis, véase su artículo.

## Motivación y objetivos

El objetivo principal de este trabajo es mejorar la calidad visual de la coloración en fractales generados mediante tiempo de escape, evitando la aparición de patrones cromáticos repetitivos incluso en zooms muy profundos. La intención es generar un conjunto potencialmente ilimitado de variaciones de color que mantengan coherencia con la paleta base elegida por el usuario, a la vez que se preservan transiciones suaves y libres de discontinuidades perceptibles.



Tiempo de escape



Tiempo de escape con algoritmo de paleta extendida

Para ello se propone un método que combina rotaciones progresivas del tono con interpolaciones cuidadosamente estructuradas, de modo que la paleta evoluciona de manera natural sin perder su identidad original.

## Algoritmo

El **algoritmo de paleta extendida** se basa en el uso de la iteración de escape suavizada y en la aplicación repetida del ángulo áureo sobre el tono (Hue) de los colores de la paleta base. Esta combinación permite generar una evolución cromática continua, coherente y libre de periodicidad.

El ángulo áureo aplicado al tono se define como:

$$\varphi_{hue} = 360^\circ \cdot \left(1 - \frac{1}{\varphi}\right), \quad \varphi = \frac{1 + \sqrt{5}}{2}$$

$$\varphi_{hue} \approx 137,50776405003785^\circ$$

Este valor es incommensurable con  $360^\circ$ , lo que significa que no existe un número entero de rotaciones que haga coincidir de nuevo el tono con su posición inicial. Como consecuencia, cada rotación sucesiva genera una variación distinta y no repetitiva del color. Esta propiedad es esencial para evitar ciclos perceptibles durante exploraciones de zoom profundo.

A continuación se describen los pasos principales del algoritmo.

- **Generación dinámica de la paleta extendida**

En lugar de construir una paleta extendida fija, los colores se generan dinámicamente cada vez que se necesitan. Para ello, cada color de la paleta base se transforma aplicando una rotación del tono (Hue) proporcional al valor entero  $k$  obtenido a partir de la iteración suavizada.

Dado un color cuyo tono original es  $H$ , la rotación se define como:

$$H' = (H + k \cdot \varphi_{hue}) \bmod 360^\circ$$

donde:

- $H$  es el tono original del color.
- $k$  proviene de la descomposición del valor de iteración suavizada (se detalla en la siguiente sección).
- $\varphi_{hue} \approx 137,50776405003785^\circ$  es el ángulo áureo aplicado sobre el tono.

Como consecuencia de esta rotación acumulativa, se obtiene una sucesión no periódica y potencialmente ilimitada de versiones cromáticas relacionadas entre sí.

- **Asignación de color mediante iteración suavizada**

Para cada punto del fractal se calcula un valor de iteración de escape suavizada  $\nu$ . Como la paleta base contiene  $L$  colores, se normaliza  $\nu$  dividiéndolo entre  $L$ , de modo que pueda interpretarse en bloques de ese tamaño.

$$\frac{\nu}{L} = k + f$$

donde:

- $k = \lfloor \frac{\nu}{L} \rfloor$  es la parte entera de  $\nu$ , y determina la rotación acumulada que se aplicará al tono de los colores base.
- $f = \frac{\nu}{L} - k$  es la parte fraccionaria de  $\nu$ , que indica la posición continua dentro del bloque y controla la transición progresiva entre la paleta base y su versión rotada.

El uso conjunto de  $k$  y  $f$  permite que la coloración evolucione sin saltos:

- $k \in \mathbb{Z}$  determina cuántas rotaciones sucesivas del ángulo áureo deben aplicarse.
- $f \in [0,1)$  regula la mezcla suave dentro de cada bloque de la paleta.

A partir de esta descomposición, el color final se obtiene siguiendo los pasos que se describen a continuación.

### 1. Selección del par de colores de la paleta base

La posición continua dentro del bloque, representada por  $f$ , determina qué dos colores consecutivos de la paleta base intervendrán en la interpolación final. Para obtener esta posición dentro de la paleta se calcula:

$$x = f \cdot L$$

donde  $L$  es el número de colores de la paleta.

A partir de  $x$  se obtienen:

$$i = \lfloor x \rfloor$$

$$\alpha = x - i$$

donde:

- $i$  es la parte entera de  $x$ , y determina el índice del color inferior del par.
- $\alpha$  es la parte fraccionaria de  $x$ , que indica la proporción de mezcla hacia el siguiente color  $i + 1$ .

De este modo quedan definidos los dos colores consecutivos de la paleta base que se interpolarán, así como la fracción exacta con la que deben mezclarse.

Es importante no confundir este parámetro  $\alpha$ , utilizado para interpolar entre estos dos colores consecutivos, con el valor  $f$ , que se empleará más adelante para mezclar cada color base con su correspondiente versión rotada.

## 2. Generación dinámica del color rotado

Una vez seleccionados los dos colores base  $B_i$  y  $B_{i+1}$ , se obtiene su versión extendida aplicando una rotación del tono proporcional al valor entero  $k$  calculado previamente. La rotación acumulada viene dada por:

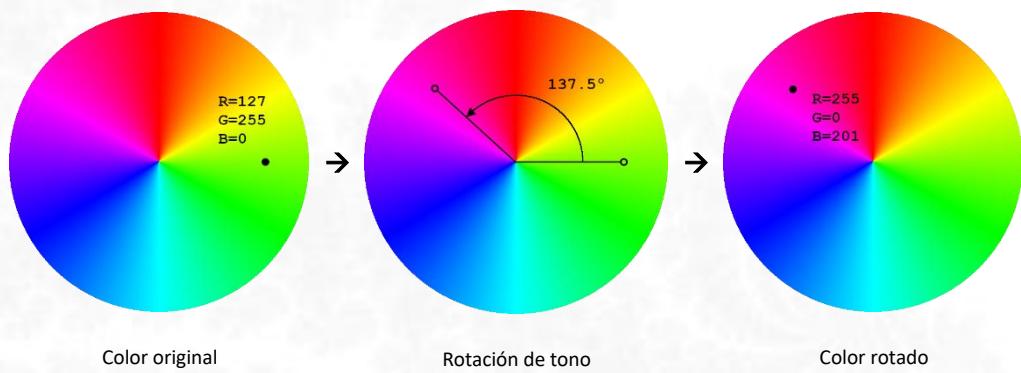
$$\theta_k = k \cdot \varphi_{hue}$$

Cada uno de los colores base se transforma entonces de la siguiente manera:

1. Se convierte el color de RGB a HSL.
2. Se rota su tono aplicando la transformación:

$$H' = (H + \theta_k) \bmod 360^\circ$$

3. Se convierte de nuevo a RGB para obtener los colores rotados  $E_i$  y  $E_{i+1}$ .



Esta rotación acumulativa genera una secuencia no periódica de variaciones de color, evitando repeticiones visibles incluso tras múltiples incrementos de  $k$ .

### 3. Mezcla entre los colores base y extendidos

Cada color base  $B_i$  y  $B_{i+1}$  se combina con su correspondiente versión rotada  $E_i$  y  $E_{i+1}$ . Esta mezcla está controlada por el valor fraccionario  $f$ , que determina el grado de transición entre la paleta original y la paleta extendida dentro de cada bloque.

La mezcla resultante se define como:

$$\begin{aligned}M_i &= (1 - f) B_i + f E_i \\M_{i+1} &= (1 - f) B_{i+1} + f E_{i+1}\end{aligned}$$

donde:

- $M_i$  es la versión interpolada del color de índice  $i$ ,
- $M_{i+1}$  es la versión interpolada del color de índice  $i + 1$ .

Estos valores  $M_i$  y  $M_{i+1}$  constituyen los dos colores entre los que se realizará la interpolación final en el siguiente paso del algoritmo.

### 4. Interpolación final entre los dos colores mezclados

La parte fraccionaria  $\alpha$ , calculada previamente, determina la posición exacta entre los dos colores ya mezclados  $M_i$  y  $M_{i+1}$ . El color final se obtiene mediante una interpolación lineal entre ambos:

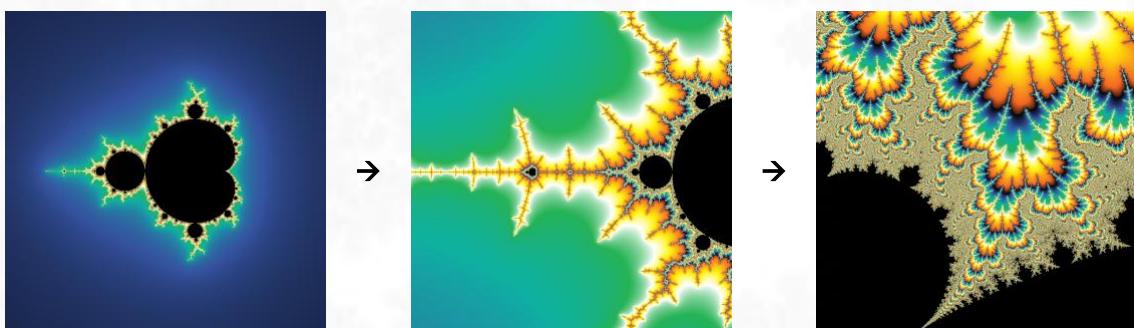
$$C(v) = (1 - \alpha) M_i + \alpha M_{i+1}$$

Esta operación proporciona una transición suave y continua al avanzar por el bloque de la paleta, garantizando que no se produzcan saltos ni discontinuidades perceptibles. El resultado  $C(v)$  es el color final asignado al punto del fractal.

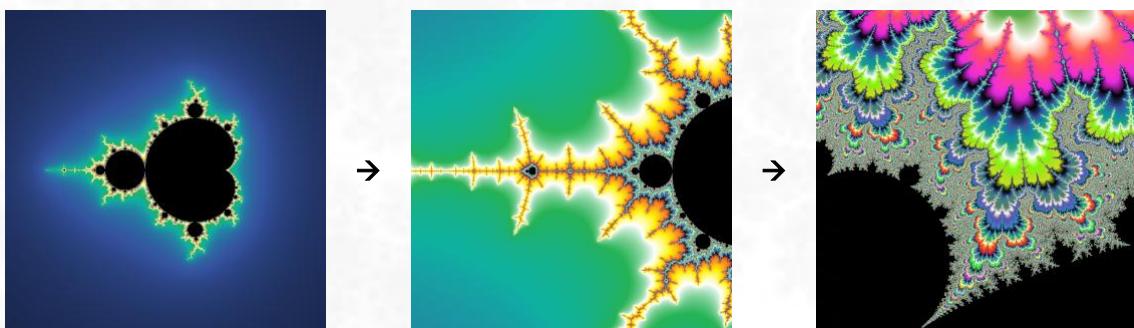
Gracias a esta construcción, el color evoluciona de forma suave con respecto a  $v$ , la transición entre bloques cromáticos permanece continua y la rotación basada en el ángulo áureo evita la aparición de patrones repetitivos incluso durante zooms muy profundos.

## Conclusiones

El método descrito permite extender la paleta original de forma continua y no periódica sin necesidad de construir una paleta extendida fija. Como la rotación aplicada al tono es proporcional al valor entero  $k$ , cuando  $k = 0$  la rotación es nula y los colores extendidos coinciden exactamente con los colores base. Esto implica que las primeras iteraciones del fractal utilizan sin alteraciones la paleta elegida por el usuario, garantizando que los colores iniciales reflejen fielmente su selección. A partir de ese momento, la transición hacia variaciones cromáticas más complejas se produce de manera gradual, suave y coherente conforme aumenta el valor de iteración suavizada.



Secuencia de zoom con paleta fija



Secuencia de zoom con paleta extendida

El algoritmo es modular y flexible, lo que facilita la experimentación con variantes como interpolaciones no lineales, rotaciones dinámicas, mezclas en otros espacios de color o gradientes definidos por el usuario, manteniendo siempre transiciones suaves y coherentes incluso durante exploraciones de zoom profundo.



Esta obra está autorizada bajo la Licencia Internacional Reconocimiento-NoComercial-CompartirIgual 4.0 de Creative Commons. Para ver una copia de esta licencia, visite <https://creativecommons.org/licenses/by-nc-sa/4.0/>