

Consejos para el Aprendiz de Brujo

# PRESTIDIGITACIÓN

Kurtian, 123RF

Dejamos nuestra GUI a un lado y editamos imágenes desde la línea de comandos. **POR BRUCE BYFIELD**

La mayoría de los usuarios creen que la edición de imágenes es una operación exclusivamente de escritorio. Después de todo, si no podemos ver los cambios que estamos realizando, ¿cómo podemos saber que son correctos? Aún así, haciendo caso omiso de esta lógica, ImageMagick [1] continúa siendo uno de los juegos de herramientas más antiguos y más extensamente utilizados para la edición de gráficos – estando su mayor eficiencia en la línea de comandos.

ImageMagick se usa muy a menudo en la edición rutinaria, donde ver los cambios no tiene importancia. Por ejemplo, puede llamarse de un script o usarse para editar un directorio de imágenes completo en una única operación en lugar de abrir trabajosamente una imagen a la vez en GIMP o Krita. Una vez nos hayamos familiarizados con ImageMagick, posiblemente encontremos que sea más rápido que sus equivalentes de escritorio para edición de imágenes, a pesar de que probablemente deseemos una herramienta de escritorio para componer una imagen original.

ImageMagick no es una aplicación en sí misma, sino un juego de herramientas multi-plataforma que funciona con más de

100 formatos de imágenes diferentes – desde formatos estándar como JPEG, GIF y PNG, a formatos sin procesar de la mayoría de tipos de cámaras digitales, pasando – con la ayuda de Ghostscript – por ficheros PDF y PostScript [2]. Generalmente se basa en la firma del formato para identificar un tipo de fichero, aunque puede recurrir al uso de la extensión si fuera necesario.

Consta de 11 comandos que comparten una estructura común y – generalmente – las mismas opciones. Estas herramientas reúnen información sobre ficheros, toman capturas de pantalla, editan gráficos y nos permiten ver gráficos en el escritorio – en resumen, todo cuanto probablemente hagamos con gráficos, incluyendo algunas tareas avanzadas que están más allá de lo que probablemente necesite el usuario medio. Aunque podemos aprender lo básico fácilmente, estos comandos y sus opciones son tan complejos, que rápidamente veremos que ImageMagick es una herramienta abierta que puede llevarnos meses o incluso años aprender rigurosamente.

## Comandos y Opciones

Como ImageMagick es fundamentalmente una aplicación de la línea de comandos, no

la encontraremos en nuestro escritorio. Tampoco encontraremos nada si introducimos ImageMagick en el prompt – en su lugar tendremos que introducir el nombre de uno de sus comandos.

La sintaxis de los comandos de ImageMagick es similar a la de la mayoría de los comandos Bash: [comando] [opciones] [fichero de entrada] [fichero de salida]. Sin embargo, debemos ser conscientes de que, aunque las opciones de ImageMagick pueden ser palabras o frases como los comandos GNU, no están prefijados por dos guiones como los comandos GNU. Sólo utilizan uno.

La selección de opciones varía con el comando. A menudo podemos usar una opción que no está relacionada con el comando para editar una imagen al vuelo. Por ejemplo, podríamos usar la opción `-size [geometría]` con el comando `composite` para cambiar las dimensiones mientras combinamos dos imágenes. En otras ocasiones, una opción no funciona con un comando, como es el caso de `-size [geometría]` y `display`. Para evitar confusiones, deberíamos eludir la edición al vuelo y hacer la mayoría de las ediciones con `convert` o `mogrify`, los dos principales comandos creados para edición detallada.

ImageMagick ofrece docenas de opciones [3], cada una de ellas con su propio juego de valores posibles. Algunas, como

```
bruce@nanday:~/work/web/first-site$ identify masthead.png
masthead.png PNG 800x140 800x140+0+0 8-bit DirectClass 49KB 0.020u
```

Figura 1: Uso del comando *identify* para leer información básica sobre un fichero imagen.

-debug y -verbose, proporcionan más información para resolver problemas y seguir la pista de lo que estamos haciendo.

La mayoría de las opciones son funciones de edición comparables a las que encontraríamos en un editor de gráficos de escritorio. Por ejemplo, podemos usar -border [geometría] y -bordercolor [color] para situar un ribete rodeando una imagen, caption [cadena] para anotarlo, -contrast para mejorar su apariencia, -crop [geometría] para cortarlo, -flip para invertir sus lados o -size [anchura]x[altura] para cambiar sus dimensiones.

ImageMagick posee incluso un número limitado de filtros para distorsionar una imagen añadiendo efectos interesantes. Por ejemplo, podemos usar -blend [porcentaje] para superponer una imagen con otra, -paint para simular una pintura al óleo o -sepia-tone para hacer que una imagen se parezca a una vieja fotografía.

Otras opciones abarcan desde aquellas que todo el mundo puede usar, a las que requieren un conocimiento profundo de la teoría del color para entender lo que pueden hacer, no hablemos de utilizarlas correctamente. Sin embargo, entender lo básico de ImageMagick puede llegar a ser simple si adoptamos la regla general de que si no entendemos una opción o sus posibles valores, no deberíamos estar usándola.

Es importante saber también que una opción puede tener diferentes efectos

```
bruce@nanday:~/work/web/first-site$ identify -verbose masthead.png
Image: masthead.png
Format: PNG (Portable Network Graphics)
Class: DirectClass
Geometry: 800x140+0+0
Resolution: 28.32x28.32
Print size: 28.2486x4.9435
Units: PixelsPerCentimeter
Type: TrueColor
Endianness: Undefined
Colorspace: RGB
Depth: 8-bit
Channel depth:
  red: 8-bit
  green: 8-bit
  blue: 8-bit
Channel statistics:
red:
  min: 0 (0)
  max: 255 (1)
  mean: 131.067 (0.513989)
  standard deviation: 115.294 (0.452135)
  kurtosis: -1.84608
  skewness: -0.0964449
green:
  min: 0 (0)
  max: 255 (1)
  mean: 124.538 (0.488383)
  standard deviation: 113.092 (0.443498)
  kurtosis: -1.82114
```

Figura 2: Si añadimos la opción *-verbose* a *identify* podemos recibir información detallada sobre un fichero gráfico, incluyendo su modelo de color.

dependiendo de la herramienta que estemos utilizando. Por ejemplo, convert -monochrome arrival.png arrival-gray.png produce una copia de una imagen en escala de grises, mientras que display -monochrome arrival.png sólo muestra la imagen en monocromo sin alterar el fichero.

### Comparación de Ficheros

Antes de editar habremos de examinar las características del fichero, así nos aseguramos de que vamos a procesar el correcto. La herramienta para mostrar información es identify. Tal y como sugiere su nombre, muestra información básica sobre un fichero. La manera más sencilla de utilizar identify es proporcionándole un nombre de fichero.

Por ejemplo, si introducimos identify masthead.png en el directorio archivado de mi antiguo sitio web, la salida resultante se muestra en la Figura 1. De izquierda a derecha, vemos el nombre del fichero, su formato de imagen (puesto que no todos los ficheros tendrán una extensión que los identifique), las dimensiones de la imagen, las dimensiones de los pseudo-lienzos (que en este caso vemos que no está en uso por los ceros que siguen), la profundidad del color y el tamaño de la imagen.

Muchos usuarios puede que sientan que esto es ya demasiada información. Sin embargo, si deseamos, podemos añadir la opción -verbose al comando para recibir algunas pantallas completas de información aún más detallada (véase la Figura 2), incluyendo el modelo de color, las estadísticas del canal y el modo de compresión. Alternativamente, podemos consultar estadísticas individuales; por ejemplo, identify -format '%k' masthead.png nos da el número de colores en una imagen.

Otra manera de compartir información del fichero es usando el comando compare (Figura 3), el cual puede ser utilizado si tenemos diferentes versiones de la misma imagen o keyframes en una secuencia de animación. Su estructura requiere dos ficheros de entrada y uno de salida.

Por ejemplo, si introducimos compare mast-head.old.jpg masthead.png masthead-compare.png, obtendría una imagen en la cual los píxeles que son diferentes en el segundo fichero se colorean de rojo – o un mensaje de que los ficheros son demasiado diferentes para hacer este tipo de comparaciones posible.

Si no quiero guardar la comparación, entonces el comando compare ./masthead.old.png ./masthead.old2.png x: muestra las diferencias en la pantalla. Una vez que he acabado de mirar, puedo pulsar la ventana que aparece sobre el escritorio y seleccionar Quit desde el menú para cerrarlo.

### Captura de Pantallas

Otra manera de seleccionar el material con el que trabajar es usando el comando import como una utilidad de captura de pantalla. Para ello introducimos el comando seguido del fichero en el que guardamos la captura de pantalla. Cuando pulsamos la tecla Enter, aparece un cursor parpadeante, y podemos pulsar en una ventana o bien elegir una sección de la pantalla a capturar. Si deseamos capturar la pantalla completa, entonces el comando es import -window root capture.png. Si necesitáramos una demora para conseguir que el terminal abandone, podemos añadir la opción -delay [segundos] inmediatamente después del comando.

### Comandos de Edición

Los comandos de edición básicos en ImageMagick son convert y mogrify. Originalmente se pretendió que convert se encargara de la producción de copias de ficheros en otro formato, pero gradualmente ha sustituido a mogrify como el comando de edición preferido.

Sospecho que la razón es que convert produce copias, requiriendo que especifiquemos tanto los ficheros de entrada como de salida. mogrify puede también producir copias, y para hacerlo requiere la adición de la opción format [tipo], lo cual es ligera-



Figura 3: El comando *compare* crea un fichero que muestra visualmente las diferencias píxel por píxel entre dos ficheros.



**Figura 4:** *Composite* superpone un fichero con otro segundo en un nuevo fichero. Aquí, especificando la gravedad como *southwest*, nos aseguramos de que el primer fichero esté posicionado en la esquina inferior izquierda.

mente más complicado y fácil de olvidar. Como ImageMagick no incluye una función undo, convert tiene una clara ventaja si cometemos un error. Aparte de eso, no existen razones por las que preferir un comando u otro.

Más allá de la edición básica tenemos el comando *composite*, que superpone una imagen sobre otra (Figura 4). Por ejemplo, si deseamos superponer *painting.png* con *mask.png* tendríamos que introducir *composite mask.png painting.png combined.png*, siendo el último fichero el fichero de salida. Si deseamos posicionar con más exactitud *parrot.png*, añadiríamos la opción *-gravity [valor]*, la cual toma valores como *Center*, *East* o *Southwest*.

### Interfaz Gráfica

Antes o después, cuando utilizamos ImageMagick, desearemos ver los cambios que hemos realizado. Cuando llega este momento, podemos utilizar el comando *display* para mostrar una imagen en nuestro escritorio (Figura 5). Por defecto, todos los ficheros indicados en el comando se presentan a tamaño completo en ventanas separadas, aunque podemos verlos fácilmente en una única ventana con un comando como *composite "vid:\*.png"*, el cual presenta todos los ficheros en el directorio actual que tienen la extensión PNG.

Sin embargo, esto no es todo lo que hace *display*. Pulsamos una de las ventanas que *display* crea para una única imagen y abrimos una ventana flotante con un menú (Figura 6). Este menú tiene un aspecto y comportamiento primitivo según lo que esperamos de los escritorios modernos, pero proporciona un acceso fácil a las opciones que la mayoría de los usuarios desean.

Casi cada lenguaje de programación ha implementado una interfaz ligeramente más sofisticada [4], desde *MagickWand* de C a *JMagick* de Java y *RMagick* de Ruby. Si se tiene curiosidad, puede verse un ejemplo de *PerlMagick* en *ImageMagick Studio* [5].

La mayoría de estas interfaces no están diseñadas pensando en usuarios finales, aunque todas son funcionales. Una desventaja es que, como todas las interfaces gráficas, están diseñadas para trabajar con una imagen a la vez, lo cual anula la ventaja principal de ImageMagick.

### Sólo el Principio

Como este artículo es una orientación básica a ImageMagick, he omitido algunos comandos tales como *conjure*, el cual se utiliza para ejecutar scripts adicionales [6], *stream* [7], que extrae los componentes de píxeles a un fichero; y *animate* [8] que, lógicamente, sirve para crear animaciones. También he de omitir las centenares de opciones y miles de combinaciones que se pueden crear con ellos. ImageMagick da para un libro, no para un sólo artículo.

Aún así, lo poco que he tratado debería ser suficiente para comenzar y trabajar. Cuando necesite más ayuda – y desee seguir ampliando conocimientos sobre ImageMagick – podrá conseguirlo consultando las explicaciones de sus herramientas tanto en formato man como en info. Para descubrir más sobre cada herramienta, podemos buscar información en la ayuda online.

Sin embargo, ninguna de esta información es lo bastante completa como la que ofrece el sitio web de ImageMagick, tanto sobre la estructura del comando en general [9], como sobre herramientas u opciones específicas [3]. El sitio del proyecto [10] tiene

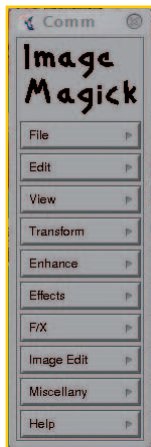


**Figura 5:** El comando *display* puede actuar como un visualizador para los ficheros gráficos del directorio actual.

también ejemplos prácticos de comandos complejos.

De manera más inmediata, si deseamos conocer los valores válidos para una opción, podemos ejecutar *-list list* para ver una lista de comandos cuyos valores se encuentren disponibles en un resumen. Luego podemos ejecutar *-list [comando]* para ver los valores posibles.

Entretanto, tome lo que desee de ImageMagick e ignore el resto. Incluso si no profundiza demasiado, encontrará una herramienta potente siempre que tenga que editar más de un fichero.



**Figura 6:** El comando *display* incluye un menú gráfico desde el cual podemos acceder a algunas de las opciones más comúnmente usadas en ImageMagick.

### RECURSOS

- [1] ImageMagick: <http://imagemagick.org/script/index.php>
- [2] Formatos de ImageMagick: <http://www.imagemagick.org/script/formats.php>
- [3] Opciones de la línea de comandos de ImageMagick: <http://www.imagemagick.org/script/command-line-options.php>
- [4] APIs ImageMagick: <http://www.imagemagick.org/script/api.php>
- [5] ImageMagick Studio: <http://www.imagemagick.org/MagickStudio/scripts/MagickStudio.cgi>
- [6] Conjure: <http://www.imagemagick.org/script/stream.php>
- [7] Stream: <http://www.imagemagick.org/script/stream.php>
- [8] Animate: <http://www.imagemagick.org/script/animate.php>
- [9] Estructura de comando de ImageMagick: <http://www.imagemagick.org/script/command-line-processing.php>
- [10] Ejemplos de ImageMagick: <http://www.imagemagick.org/Usage/>