

LoadImage by neofar

<http://ps2dev.ofcode.com>

Mediante esta utilidad será posible leer imágenes en formato TGA o BMP y convertirlas a IIF o a RAW, el programa acepta TGA en 32,24,16, y 8 bits incluidos con compresión RLE con los BMP de momento solo soportamos 24 y 8 bits, sin comprimir. Por ahora no hemos implementado ningún formato de 4 bits, por lo que no es posible guardar imágenes en CLUT4.

		TGA (RLE)				BMP		
		32	24	16	8	24	8	4
RAW	RAW32	X						
	RAW24		X			X		
	RAW16			X				
IIF	RGBA32	X						
	RGB24		X					
	RGBA16			X				
	CLUT8RGBA32				X		X	
	CLUT8RGBA16				X		X	
	CLUT4RGBA32							X
	CLUT4RGBA16							X

Usage: LoadImg.exe <filein> <fileoutput> <output_format> <XOptions>

Parámetros

filein	Input filename (TGA(+RLE) BMP)
fileoutput	File to create (RAW IIF)
output format	Format to convert
XOptions	Read documentation (this)

Formatos de salida

RAW32 RAW24 RAW16
RGBA32 RGB24 RGBA16
CLUT8-RGBA32 CLUT8-RGBA16
CLUT4-RGBA32 CLUT4-RGBA16 (not implemented)

Ejemplo

loadImg.exe image1.tga out/32.raw RAW32
loadImg.exe image2.tga out/24.iif RGB24
loadImg.exe image3.bmp out/8.iif CLUT8-RGBA32

Extended Options (XOptions)

Flipv	Invierte verticalmente la imagen
Fliph	Invierte horizontalmente la imagen
Orderpal	Ordena la paleta por luminosidad
Alpha	Fuerza un color determinado dado por RGB para que sea transparente
AlphaRamp	Crea una escala de alpha en la paleta

Xtended Options

- Flipv

Mediante este parámetro vamos a invertir la imagen verticalmente, es muy útil cuando estamos cargando una imagen en formato TGA, que por defecto se guardan invertidas.

ejemplos:

```
loadImg.exe image1.tga out/32.raw RAW32 flipv
```

- Flih

Igual que el flipv pero hace un reflejo horizontal.

ejemplos:

```
loadImg.exe image1.tga out/32.raw RAW32 fliph
```

- Orderpal

Ordena la paleta de colores por luminancia, sólo se puede ejecutar en los modos CLUT. Si queremos hacer desde la PS2 cualquier manipulación con la paleta (rotaciones de saturación) será necesario que la paleta se encuentre ordenada por luminancia. Si queremos invertir el orden de la paleta deberemos poner 2 veces el parámetro

ejemplos:

```
loadImg.exe image3.bmp out/8.iif CLUT8-RGBA32 orderpal
```

```
loadImg.exe image3.bmp out/8.iif CLUT8-RGBA32 orderpal orderpal
```

- Alpha <r> <g>

Mediante este comando podremos hacer que un color cualquiera de la paleta sea transparente, normalmente se usa el fusia para indicar el alpha en las imágenes en 8 bits, sabiendo que el fusia es el color R=255 G=0 B=255 podríamos hacerlo transparente mediante el comando 'alpha 255 0 255'.

Este comando se puede usar varias veces en la misma línea de comandos para indicar varios colores como transparentes

ejemplos:

```
loadImg.exe image3.bmp out/8.iif CLUT8-RGBA32 alpha 255 0 255
```

```
loadImg.exe image3.bmp out/8.iif CLUT8-RGBA32 alpha 255 0 255 alpha 0 0 0
```

- AlphaRamp <i1> <c1> <i2> <c2>

Este comando es una extensión del alpha, en lugar de crear un color como transparente crea una escala de transparencia, debemos darle 4 parámetros:

i1 -> posición inicial en la paleta

c1 -> valor alpha inicial

i2 -> posición final de la paleta

c2 -> valor alpha final

ejemplos:

```
loadImg.exe image3.bmp out/8.iif CLUT8-RGBA32 alpharamp 0 0 16 255
```

Este ejemplo le dice que los colores del 0 al 16 tendrán un alpha que va desde 0 a 255, ósea.. los 16 primeros colores de la paleta tendrán un suave degradado con el alpha

```
loadImg.exe image3.bmp out/8.iif CLUT8-RGBA32 alpharamp 0 0 16 255 alpharamp 128 255 255 0
```