

Kara Effector 3.2:

Effector Book

Vol. II [Tomo XXVIII]



Kara Effector 3.2:

En este **Tomo XXVIII** continuaremos viendo las funciones de esta interesante **librería text**. Esta librería contiene una serie de funciones interesantes que espero que con la ayuda de esta documentación, le puedan sacar el máximo provecho a la hora de llevar a cabo sus proyectos, no solo karaokes, sino también en la edición de los subtítulos.

Librería Text [KE]:

» `text.to_clip(Text, relative_pos, iclip)`

Esta función retorna un clip con la forma exacta del texto ingresado. El parámetro **Text** es el string de texto a ser convertido en clip y su valor por default es el texto por default según el **Template Type** del efecto aplicado.

Los parámetros **relative_pos** e **iclip** son dos booleanos, el primero hace que el clip generado tenga la misma posición del texto original sin importar cuál sea su posición inicial y su valor por default es **nil**. El parámetro **iclip** decide si el texto ingresado se convertirá en un clip o en un iclip. Su valor por default es **nil**.

» Ejemplo:

Para este ejemplo usamos un **Template Type: Word**

Template Type [fx]:	Word
Add Tags Language:	Lua
<code>text.to_clip(word.text)</code>	

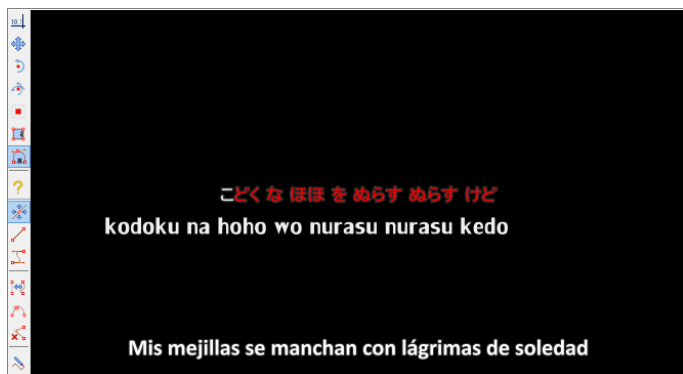
Y modificamos un poco las posiciones por default de cada Palabra generada por el efecto, por ejemplo así:

Pos in "X" =	<code>fx.pos_x - 100</code>
Pos in "Y" =	<code>fx.pos_y + 125</code>

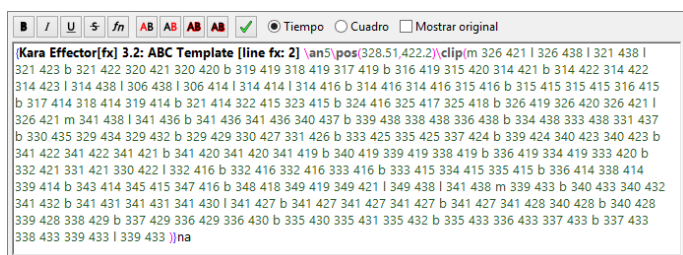
Es decir, 100 pixeles a la izquierda de su centro por default y 125 pixeles hacía abajo.

A aplicar, en apariencia solo veremos el texto en la posición que debería tener dada las modificaciones que previamente le indicamos.

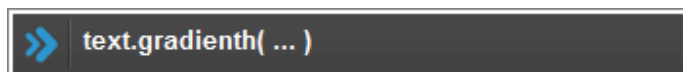
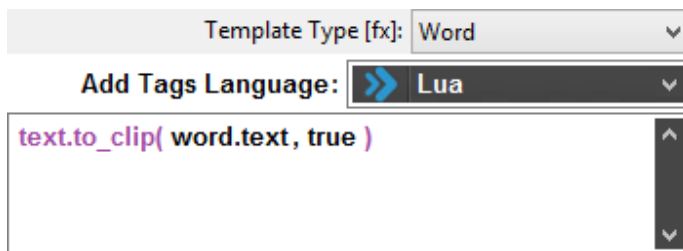
Y vemos que los clip's son generados en la posición en la que se encuentra el texto sin importar que éste no esté en su posición por default: (**word.text**, **word.middle**)



Para poder apreciar el clip que genera esta función, es necesario que veamos en detalle una de las líneas fx generadas:



Si queremos que el clip generado tenga la posición por default del texto sin importar las modificaciones en las posiciones, ponemos true en el segundo parámetro de la función:

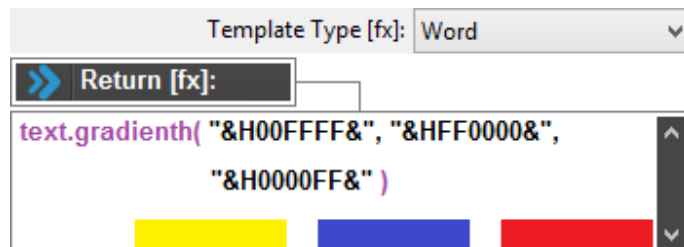


Crea un gradiente (degradado) horizontal entre todos los colores ingresados, sin la necesidad de usar el **VSTFilterMod** y solo generando una única línea de fx.

La función convierte el texto por default según el **Template Type** del efecto, en un clip y luego genera dentro de él una serie de Shapes consecutivas de manera cada una tenga un color diferente y entre todas juntas conformen en gradiente horizontal.

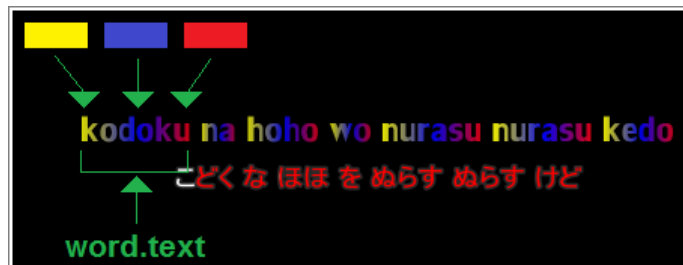
Ejemplo:

Para este ejemplo usaremos un **Template Type: Word**, y pondremos tres colores en la función, pero la cantidad total no tiene límites:



Recordemos que para crear un gradiente horizontal o vertical, la mínima cantidad de colores a ingresar en la función es de dos.

Al aplicar el ejemplo, notamos cómo cada palabra (Word) contiene un gradiente entre los tres colores ingresados:



Si ampliamos un poco la imagen veremos cómo la función crea el gradiente entre los tres colores, en cada palabra de la línea karaoke:



Una de las ventajas de esta función es que crea el gradiente horizontal entre todos los colores ingresados, generando una única línea de fx, en este ejemplo, por cada palabra de cada línea a la que se le aplica el efecto:

25	3	0:00:45.92	0:00:54.56	English				Dos corazones que se buscan cor
26		0:00:02.43	0:00:08.16	Romaji	lead-in	Effector [Fx]	*m 0 0 1 0 7 4 1 2 7 4 1 2 0 1 0 0	*m
27		0:00:02.43	0:00:08.16	Romaji	lead-in	Effector [Fx]	*m 0 0 1 0 7 4 1 2 7 4 1 2 0 1 0 0	*m
28		0:00:02.43	0:00:08.16	Romaji	lead-in	Effector [Fx]	*m 0 0 1 0 7 4 1 2 7 4 1 2 0 1 0 0	*m
29		0:00:02.43	0:00:08.16	Romaji	lead-in	Effector [Fx]	*m 0 0 1 0 7 4 1 2 7 4 1 2 0 1 0 0	*m
30		0:00:02.43	0:00:08.16	Romaji	lead-in	Effector [Fx]	*m 0 0 1 0 7 4 1 2 7 4 1 2 0 1 0 0	*m
31		0:00:02.43	0:00:08.16	Romaji	lead-in	Effector [Fx]	*m 0 0 1 0 7 4 1 2 7 4 1 2 0 1 0 0	*m
32		0:00:02.43	0:00:08.16	Romaji	lead-in	Effector [Fx]	*m 0 0 1 0 7 4 1 2 7 4 1 2 0 1 0 0	*m
33		0:00:08.33	0:00:13.19	Romaji	lead-in	Effector [Fx]	*m 0 0 1 0 7 4 1 2 7 4 1 2 0 1 0 0	*m
34		0:00:08.33	0:00:13.19	Romaji	lead-in	Effector [Fx]	*m 0 0 1 0 7 4 1 2 7 4 1 2 0 1 0 0	*m

Esta ventaja hace que el **script** sea más liviano y sea leído fácilmente, en el caso de los subtítulos en modo **"Softsub"**.

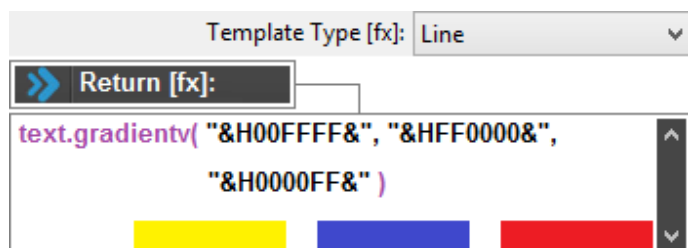
text.gradientv(...)

Crea un gradiente (degradado) vertical entre todos los colores ingresados, de manera similar a la función anterior, sin la necesidad de usar el **VSTFilterMod** y solo generando una única línea de fx.

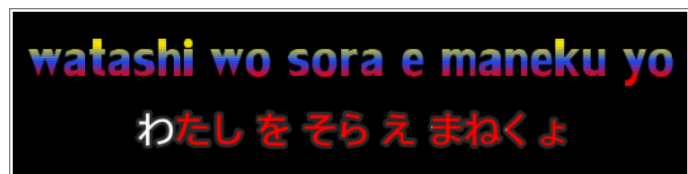
La función convierte el texto por default según el **Template Type** del efecto, en un clip y luego genera dentro de él una serie de Shapes consecutivas de manera cada una tenga un color diferente y entre todas juntas conformen en gradiente vertical.

Ejemplo:

Con un **Template Type: Line**, y en **Return [fx]** escribimos lo siguiente:



Y al aplicar:



Aunque aún restan varias funciones de la **librería text** por documentar, por ahora le daré prioridad a otras librerías, funciones y herramientas, que como todas las anteriores, nos ayudarán a sacarle un mayor provecho al **Kara Effector** y por ende a todos nuestros proyectos desarrollados en él.

Siguiendo con este orden de ideas, la siguiente librería en documentar será la **librería effector**.

Librería Effector [KE]:

Esta librería contiene un listado de funciones especiales, las cuales la mayoría son "**funciones administrativas**", ya que son funciones internas del **KE** que no generan efectos sino que hacen que él realice su tarea de forma correcta.

No vale la pena el entrar en detalle con las anteriormente mencionadas funciones administrativas, pero al igual las mencionaré. Por otra parte, hay unas cuantas funciones en esta librería que sí nos pueden ser de gran utilidad.

El listado de funciones que contiene la **librería effector** es el siguiente:

1. **effector.pos**
2. **effector.knj**
3. **effector.offset_pos**
4. **effector.import**
5. **effector.addfx**
6. **effector.savefx**
7. **effector.modify_pos**
8. **effector.new_pos**
9. **effector.default_val**
10. **effector.effect_offset**
11. **effector.decide**
12. **effector.print_error**
13. **effector.run_fx**
14. **effector.preprocesses_styles**
15. **effector.preprocesses_macro**
16. **effector.preprocesses_lines**
17. **effector.macro_fx**

De las anteriores funciones, las que están en negro son las funciones administrativas, y las que están en azul son las que de una u otra manera le podemos sacar provecho.

effector.offset_pos()

Es una función interna del **KE** que hace que las líneas de efecto generadas queden posicionadas de forma relativa a una posición previa hecha a las líneas del script en el vídeo.

Ejemplo:

Lo que debemos hacer es seleccionar la o las líneas que queremos reposicionar en el vídeo, para ello usaremos la herramienta para arrastrar los subtítulos del **Aegisub** que está en la esquina superior izquierda, al lado del vídeo:



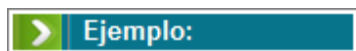
El procedimiento es simple, ya que una vez que pulsamos el botón para arrastrar los subtítulos, a cada elemento visible en el vídeo le sale un cuadrado con una cruz al centro que indica la posición y alineación del mismo, y con el clic derecho del mouse en él elegimos la nueva posición:



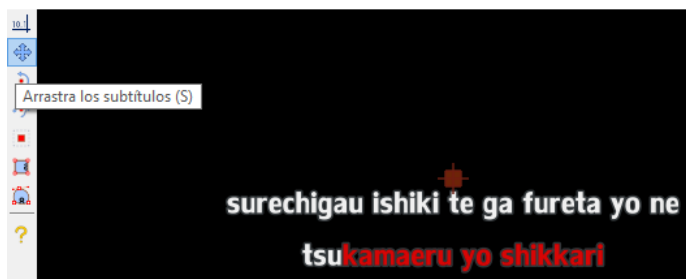
Una vez que hayamos desplazado la línea seleccionada a una nueva posición, la función **effector.offset_pos** hará internamente que esas coordenadas sean el punto de referencia al aplicar un efecto. Ejemplo:



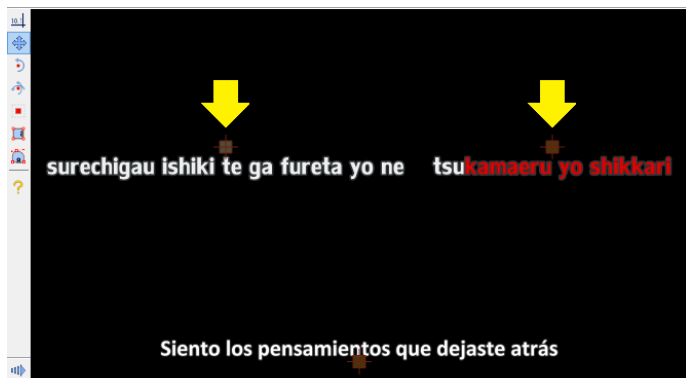
Esta habilidad interna del **KE** es ideal, por ejemplo, para aquellas líneas que coinciden en tiempo y se superponen en el vídeo:



Las líneas tienen el mismo centro, y al coincidir en tiempo, se superponen:



Entonces podemos mover las líneas de forma que ya no coincidan en su posición:

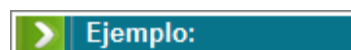


Y una vez que ya le hemos dado nuevas posiciones a las líneas en conflicto, ya podemos aplicar un efecto sin el temor de que las líneas de fx vayan a quedar superpuestas:

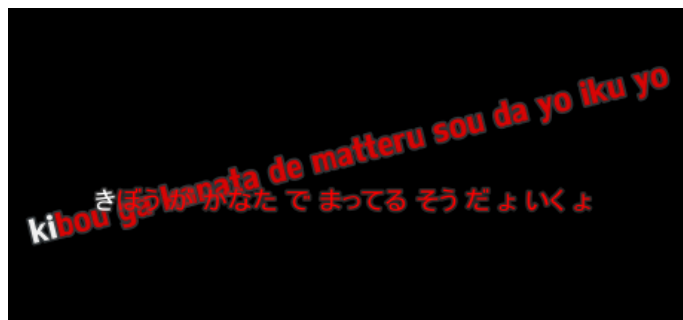


En resumen hasta acá, esta función hace que el **KE** redefina los centros que por default ya tiene cada línea del script dados por los valores de posición y alineación en el estilo de cada línea.

Otra habilidad que tiene esta función interna del **KE** es poder reposicionar a las líneas de karaoke previamente con la herramienta de rotación en los estilos de las mismas:



Y en vídeo veremos algo como esto:



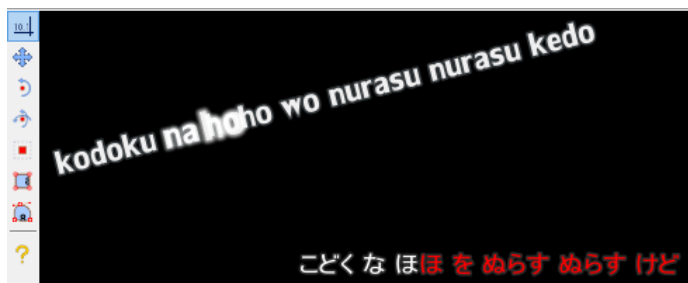
Y al aplicar los efectos, el texto completo adoptará el ángulo de inclinación puesto en los estilos:



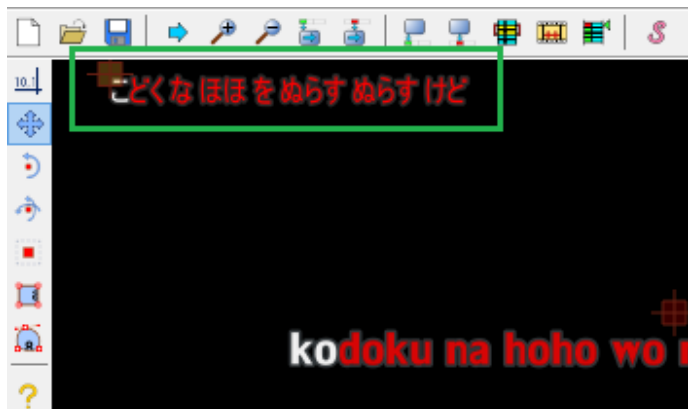
Una vez seleccionado el ángulo de inclinación de las líneas, podemos también modificar la posición en el vídeo con la herramienta de arrastre de subtítulos:



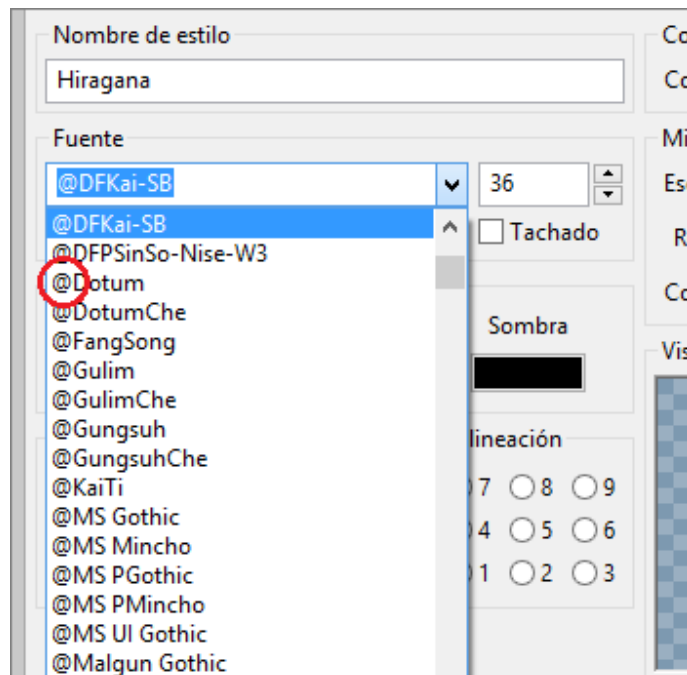
Y al aplicar los efectos seleccionados, la función reubicará automáticamente al texto para que las posiciones y ángulos de inclinación sean los mismos:



Otra de las utilidades de esta habilidad se usa para aplicar efectos a los **kanjis**, pero de forma vertical y utilizando las **fonts** que ya vienen prediseñadas para que el **kanji** se vea vertical con solo modificar la rotación del texto:



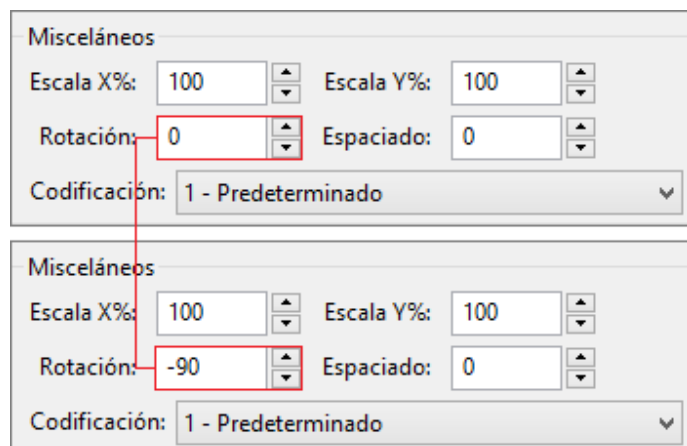
Las **fonts** que vienen prediseñadas para los **kanjis** en vertical, son la que en su nombre pone el símbolo de arroba (@) al inicio de su nombre:



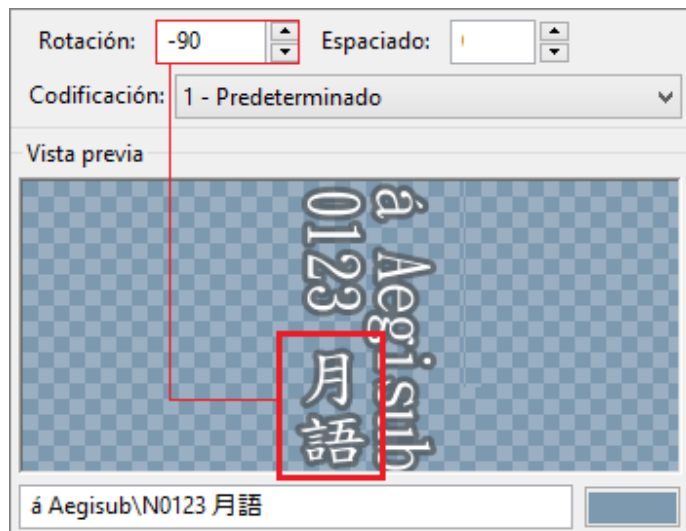
Seleccionamos una de esas **fonts** que más nos guste y se ajuste a nuestro proyecto, y vemos cómo el texto **kanji** está rotado en un ángulo de 90°:



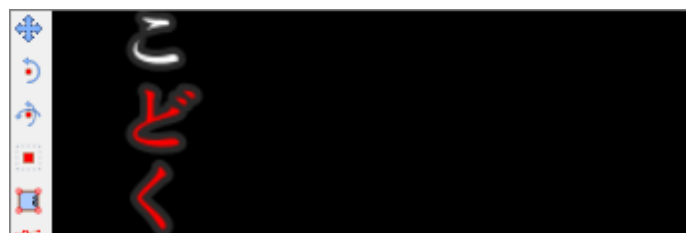
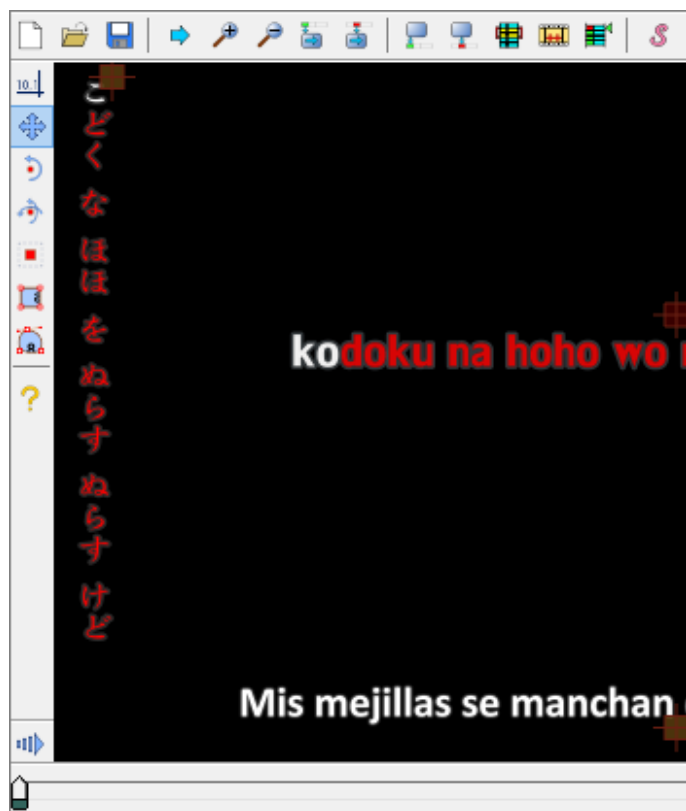
Entonces modificamos la rotación del texto así:



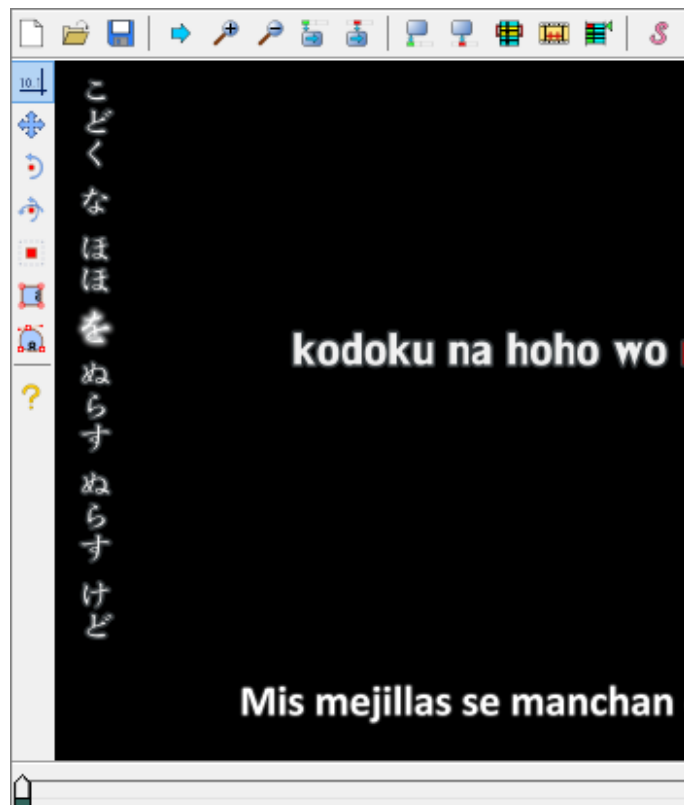
Ahora el **kanji** ya se ve bien en vertical aunque el resto del texto esté rotado -90°. Este hecho en particular no importa siempre y cuando en las líneas de texto solo haya **kanji**:



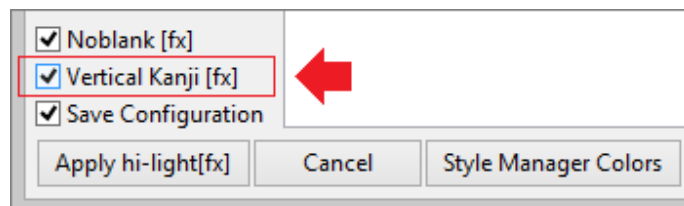
Hecho esto ya podemos ver en el vídeo cómo se ve el estilo del **kanji** en vertical sin aún haber aplicado ningún efecto:



Aplicando todos los efectos, la función hace los ajustes necesarios para que el texto **kanji** quede de forma vertical usando las **fonts** prediseñadas para ello:



La ventaja de esta habilidad es que hay un listado grande de **fonts** para **kanji** en vertical, y que de otro modo no podemos hacer con las **fonts** que son diseñadas para texto horizontal. Recordemos que para aquellas **fonts** de texto horizontal usamos la opción de "Vertical Kanji [fx]" del KE y obtenemos los mismos resultados:



Es todo por ahora para el **Tomo XXVIII**. Intenten poner en práctica todos los ejemplos vistos y no olviden descargar la última actualización disponible del **Kara Effector 3.2** y visitarnos en el **Blog Oficial**, lo mismo que en los canales de **YouTube** para descargar los nuevos Efectos o dejar algún comentario. Pueden visitarnos y dejar su comentario en nuestra página de **Facebook**:

- www.karaeffector.blogspot.com
- www.facebook.com/karaeffector
- www.youtube.com/user/victor8607
- www.youtube.com/user/NatsuoKE
- www.youtube.com/user/karalaura2012