

Hyperdimensional Relocator Tutorial (Simplified Chinese)

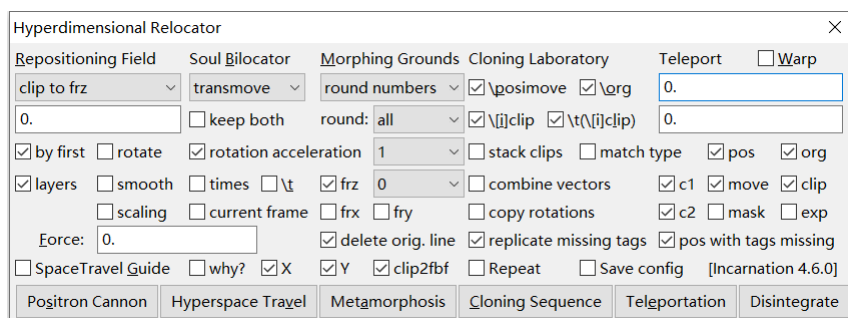
此篇教程将会对 Aegisub 插件 Hyperdimensional Relocator 的官方英文介绍进行翻译，并额外对各项操作进行解释。

教程测试环境: Aegisub 3.2.2

插件版本: 4.6

译者: ihkk

手册版本: 4.6 Alpha



官方文档对照翻译

介绍

原文	译文
<p>Introduction</p> <p>Hyperdimensional Relocator offers a plethora of functions,focusing primarily on \pos, \move, \org, \clip, and rotations. Anything related to positioning, movement, changing shape, etc.,Relocator aims to make it happen.</p> <p>Position Cannon deals mostly with \pos, Hyperspace Travel with \move, Cloning Sequence copies from line to line, Teleportation changes coordinates, and Metamorphosis changes various things.</p> <p>Current version: 4.6.0</p> <p>Update location: http://unanimated.hostfree.pw/ts/relocator.lua</p>	<p>介绍</p> <p>Hyperdimensional Relocator 提供诸多功能，主要聚焦 \pos, \move, \org, \clip 和旋转（译者注：\frz）</p> <p>此插件旨在实现各种位置、移动、变形等相关变化。</p> <p>Position Cannon（将译作 定位区）主要处理 \pos 有关的变化，Hyperspace Travel（将译作 位移区）主要处理 \move 有关的变化，Cloning Sequence（将译作 克隆区）可以克隆多条字幕，Teleportation（将译作 坐标移动区）可以更改坐标，Metamorphosis (将译作 变形区)可以处理很多东西（译者注：???）。</p> <p>当前版本：4.6.0</p> <p>更新地址：http://unanimated.hostfree.pw/ts/relocator.lua</p>

Position Cannon 定位区

Repositioning Field

clip to frz

Align X

Align Y

org to fax

clip to fax

clip to frz

clip to reposition

clip2pos fbf

frz+org2pos

horizontal mirror

vertical mirror

numbers

shake

fbf retrack

trampoline

sine loop

shadow layer

shadow repos.

space out letters

warp text

replicate...

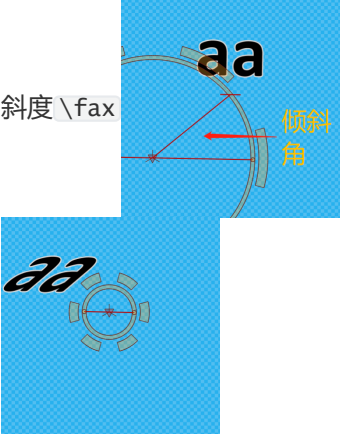
fbf X <--> Y

track by clip

原文	译文
<p>Align X means all selected \pos tags will have the same given X coordinate. Same with Align Y for Y.</p> <p>Useful for aligning multiple signs horizontally/vertically or mocha signs that should move horizontally/vertically.</p> <p>'by first' aligns by X or Y from the first line.</p>	<p>Align X，可以将所有的被选中的 \pos 标签的X轴坐标统一。Align Y同理，用于统一Y轴坐标。</p> <p><i>此选项对于使多个标签水平竖直很有用，也可以在Mocha追踪时使用。对于一些Mocha追踪的结果，可以将原本应该是水平/竖直移动但跟踪错误的追踪轨迹校准。</i></p>
<p>Horizontal Mirror : Duplicates the line and places it horizontally across the screen, mirrored around the middle. If you input a number, it will mirror around that coordinate instead,</p> <p>so if you have \pos(300,200) and input is 400, the mirrored result will be \pos(500,200).</p> <p>'keep both' will keep the original line along with the mirrored one. 'rotate' will flip the text accordingly.</p> <p>Vertical Mirror is the logical vertical counterpart.</p>	<p>Horizontal Mirror，可以将选中的字幕的位置关于屏幕中心轴对称（默认不保留初始字幕）。如果你在下方的输入框中填入了数字，那么就会以输入的直线X=?作为对称轴。例如要变换的坐标是 \pos(300,200) 而输入的对称轴是 400，那么镜像得到的便是 \pos(500,200)。</p> <p>下方有两个选项，keep both 和 rotate。如果勾选了 keep both，那么对称变换会保留初始的字幕。如果勾选了 rotate，那么便会旋转相应的度数，此处为180°。</p>

原文	译文
<p>Shake: Apply to fbf lines with \pos tags to create a shaking effect.</p> <p>Input radius for how many pixels the sign may deflect from the original position.</p> <p>(Use Teleport coordinates if you want different values for X and Y.)</p> <p>'scaling' randomises fscx/y. Value '6' with fscx150 -> 144-156. (Uses input from the Force field.)</p> <p>'rotate' adds shaking to \frz (pos input, degrees). 'smooth' will make shaking smoother.</p> <p>'layers' will keep position/rotations the same for all layers. (Same value for same start time.)</p>	<p>Shake 抖动效果：需要应用于有 \pos 标签的 fbf (frame-by-frame, 被逐帧拆开的字幕) 多条字幕。</p> <p>在框中输入想要每次随机偏移的x&y范围 (译者注：例如，原先的坐标为 \pos(100,100)，填入的范围为 10，那么随机生成的位置将使 \pos(100±10,100±10)。</p> <p>(特别的，你可以在 Teleport坐标移动区 (右上角) 分别填入想要偏移的x和y的范围。)</p> <p>勾选 scaling，并在下方的 Force 输入框中，键入变形的范围，可以在抖动的同时，加入变形效果。例如键入了 6，而原先的变形是 \fscx150，那么就会随机变形为 \fscx144-\fscx156。</p> <p>勾选 rotate，可以在抖动时应用随机的旋转效果。旋转的角度为 \frz±填入的xy位移范围。</p> <p>(默认) 勾选 layers，会使所有选中的字幕进行统一的随机移动。(相同开始时间有相同的抖动) (译者注：此选项仅在同时多句字幕时有效)</p>
<p>Trampoline / Sine Loop: Creates fbf movement between point A and point B. Check online manual for details.</p>	<p>Trampoline / Sine Loop：生成两点间的 fbf逐帧字幕。更多参考网上手册。</p> <p>译者注：Trampoline 为抛物线，Sine 为正弦曲线。这两个功能都可以实现一句字幕在两点之间来回运动的效果。使用时，应在 Teleport坐标移动区 (右上角) 分别填入运动终点的x和y的偏移量。</p>

原文	译文
<p>Shadow Layer: Creates shadow as a new layer. For offset, it uses in this order of priority:</p> <p>1. value from Positron or Teleport (xshad, yshad). 2. shadow value from the line. 3. shadow from style.</p>	<p>Shadow Layer: 创建一个新的纯阴影字幕，将原字幕和其阴影分离开来。</p> <p>对于阴影的偏离值，遵从以下优先级：</p> <p>1. 左上角的 disPosition 输入框填入的数值，或右上角的 Teleport 输入框填入的 xshad 和 yshad 数值。</p> <p>2. 字幕中标签的数值。</p> <p>3. 字幕样式中的阴影数值。</p>
<p>Shadow Reposition: "centers" text with shadow by the \shad distance. With \shad4, moves 2 pixels left & up. Works with [xy]shad. Useful when top layer has shadow & needs centering within bottom layer with border.</p>	<p>Shadow Reposition: 重新定位阴影位置。根据阴影距离，“居中”字幕。例如阴影长度为 \shad4，那么会向左上方移动 2 个像素。</p> <p>适用于 \[xy]shad。在需要对齐多个图层时可能有用。（译者注：不太懂）</p>
<p>fbf X <--> Y: The fbf increase/decrease in pos X will become that of pos Y and vice versa.</p> <p>If 'by first' is checked, the first line is the reference line (won't change). Otherwise, it's the last line.</p> <p>With 'rotate' checked, the resulting direction is reversed. (Recalculator's Mirror adds more options to this.)</p> <p>Example: If a sign moves 100 pixels to the right, with 'first' it will move 100px down (or up with 'rotate').</p> <p>Without 'first', the directions are the same, but sign will end where it did before, not start.</p>	<p>fbf X <--> Y: 将 fbf 逐帧字幕中 \pos 位置代码里的 x 轴变化量和 y 轴变化量互换。</p> <p>如果勾选 by first，那么就以第一句作为基准，不会改变；如果不勾选，就以最后一句作为基准不改变。</p> <p>如果 rotate 变勾选，那么运动方向将会相反。（译者注：例如原先运动方向是 →，勾选了 rotate 则会变为 ↑，不勾选则是 ↓。）</p> <p>例如：如果原先是向右移动 100 像素，勾选了 by first 就会向下移动 100 像素（如果勾选 rotate 则会向上移动），如果没勾选 by first，运动方向依然是向下，但原先的运动起点将会是新的运动的终点。）</p>
<p>Numbers: Informative only function that shows \pos differences between consecutive selected lines.</p> <p>Can be useful if you want to continue a fbf movement and need an idea of the distances required.</p>	<p>Numbers: 对于选中的连续字幕，可以显示 \pos 的变换量。</p> <p>在你需要补充连续移动的 fbf 逐帧字幕时，可以列表显示一次需要移动多少量。</p>

原文	译文
<p><code>org to fax</code>: Calculates <code>\fax</code> from the line between <code>\pos</code> and <code>\org</code> coordinates.</p>	<p><code>org to fax</code>: 在绕Z轴旋转的可视化操作界面, 通过改变字幕和中心的位置, 来调整倾</p> 
<p><code>clip to fax</code>: Calculates <code>\fax</code> from the line between the first 2 points of a vectorial clip. Both of these work with <code>\fscx</code>, <code>\fscy</code>, and <code>\frz</code>. If the clip has 4 points, points 3-4 are used to calculate <code>fax</code> for the last character, and a gradient is applied. See blog post for more info - http://unanimated.hostfree.pw/itw/tsblok.htm#fax</p>	<p><code>clip to fax</code>: 与 <code>org to fax</code> 类似, 只是倾斜角为 <code>\clip</code> 中两点连线的倾角。如果 <code>\clip</code> 有四个点, 那么第3-4个将会被作为参考点。</p> 
<p><code>Clip to Frz</code>: Calculates <code>\frz</code> from the first 2 points of a vectorial clip. Direction of text is from point 1 to point 2. If the clip has 4 points, the <code>frz</code> is average from 1-2 and 3-4. (Both lines must be in the same direction.) This helps when the sign is in the middle of a rectangular area where the top and bottom lines converge.</p>	<p><code>clip to frz</code>: 与 <code>clip to fax</code> 类似, <code>\clip</code> 直线倾斜角为旋转的度数。如果 <code>\clip</code> 有四个点, 那么 <code>\frz</code> 会选取1-2和3-4的平均值。(二者需要平行)</p>
<p><code>clip to reposition</code>: Shifts <code>\pos</code> based on first 2 points of a vectorial clip. You can use a reference point in the video image and set the clip points at start/end of the movement.</p>	<p><code>clip to reposition</code>: 根据 <code>\clip</code> 中两点的相对位置, 移动字幕。你可以结合视频中的参考点来使用。</p>
<p><code>clip2pos fbf</code>: This is for manual fbf tracking. Track something in the video with one point of a clip in each frame. The first line is a reference point. The <code>\pos</code> of other lines will shift based on the difference between the clip on the first line and the clip on the current line. (Starting point is always <code>\pos</code> from line 1.)</p>	<p><code>clip2pos fbf</code>: 适用于手动追踪, 在每一帧中用一个点追踪。第一行是一个参考点, 其他行的 <code>\pos</code> 将会在第一行的基础上进行移动。</p>

原文	译文
<p><code>frz+org2pos</code>：把有 <code>\org</code> 的旋转变为无 <code>\org</code> 的旋转，显示效果不变。</p>	
<p><code>space out letters</code>：把一句字幕的所有字符拆开，分散在 <code>\clip</code> 的选取中。</p>	
<p><code>warp text</code>：使字幕向上/向下拱起。需要配合 <code>space out letters</code> 使用，先分离文字，然后在左上角框中输入拱起的值，正为上负为下。</p>	
<p><code>replicate</code>：重复字幕，并且可以设置单词重复的位移量。</p> <p>如右图：<code>replicas</code>：重复次数 <code>[XY] Distance</code> 单次偏移量</p>	

原文

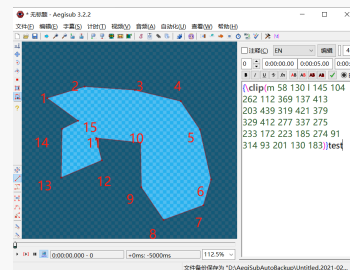
Track by clip: 根据 \clip 的轨迹移动追踪。会根据 \clip 的点的位置, 生成 fbf 逐帧字幕的 \pos 的位置

fbf retrack: There is simple mode and 'smoothen track' mode.

Simple mode is like fbf-transform for position, but it can be applied to several layers at the same time and have different accel for X and Y. If you check 'layers', the scale is not by selection but by start time. You can have subtitles sorted by layers, but each layer must be sorted by time. Of course each frame will have the same position for all layers/signs. You can use Repositioning Field for accel, or Teleport for separate X/Y accel. You can track only X or Y if you check only one of those. Without 'layers' checked, it simply goes through selected lines and can be applied to signs in the same frame.

'Smoothen track' mode is activated when you check 'smooth'. This is designed for smoothening tracking data, i.e. it will move positions that stand out closer to the main line of the track. It would make no sense to apply this to shaking signs, but if you have trouble tracking something in mocha and the sign tends to jump off on some frames, this will pull the jumps back in line. You can apply different strength of smoothening, by using the Force Field. 0 is the lowest strength; 100 is the highest and will make the track a straight line.

译文



fbf retrack: 拥有 简单模式 和 平滑模式。

简单模式 比较像 位置 \pos 的 fbf 逐帧转换, 但可以同时应用于不同行的字幕, 同时可以分别设置 x 和 y 的加速度。

(译者注: 使用条件: 已经有多条时间不同但位置相同的 fbf 字幕, 且第一句、最后一句的位置不同。) 可以勾选 layers 来区分不同的段。如果开始时间一致, 那么相同帧的不同字幕将会有相同的位置。

你可以在左上角的

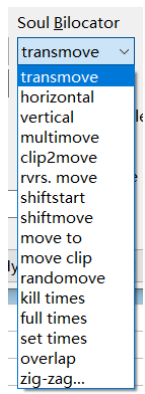
disPosition 输入框填入加速度, 或右上角的 Teleport 输入框填入独立的 x 和 y 的加速度。

平滑追踪 模式将在你勾选了 smooth 时被激活。这是用来平滑数据的, 也就是说, 它会将移动更靠近主字幕。你不应该对 抖动的 字幕应用这个选项, 但是可以用在 mocha 追踪时, 例如, 某一个帧追踪的位置突然偏离了正确的轨迹, 那么这个模式可以把脱离的字幕拉回到应在的位置。

你可以使用不同的平滑力度, 在 Force Field 中填入数值。0 是最低力度, 100 是最高数值, 会让轨迹变成一条直线。

(译者注: 可以理解为 拟合)

Hyperspace Travel 位移区

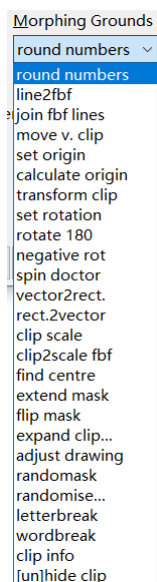


原文	译文
<p>Transmove</p> <p>Main function: create \move from two lines with \pos. Duplicate your line, and position the second one where you want the \move the end. Script will create \move from the two positions. Second line will be deleted by default; it's there just so you can comfortably set the final position. Extra function: to make this a lot more awesome, this can create transforms. The second line is used not only for \move coordinates but also for transforms. Any tag on line 2 that's different from line 1 will be used to create a transform on line 1. So for a \move with transforms, you can set the initial sign and then the final sign while everything is static.</p> <p>You can time line 2 to just the last frame. The script only uses timecodes from line 1. Text from line 2 is also ignored (assumed to be same as line 1). You can time line 2 to start after line 1 and check 'keep both.' That way line 1 transforms into line 2, and the sign stays like that for the duration of line 2.</p> <p>'Rotation acceleration' - like with fbf-transform, this ensures that transforms of rotations will go the shortest way, thus going only 4 degrees from 358 to 2 and not 356 degrees around.</p> <p>If the \pos is the same on both lines, only transforms will be applied.</p> <p>Logically, you must NOT select 2 consecutive lines when you want to run this, though you can select every other line.</p> <p>'times' will add timecodes to \move</p>	<p>Transmove：从上下两句 \pos 中创建第一句的 \move。复制你的字幕，然后重新定位第二句，使它放置于你想要 \move 结束的位置。【译者注：此时，选中第一句，而不要同时选择两句！】脚本会创建这两个位置中的 \move 语句。第二句话会被自动删除，它存在的目的只是为了让你设置移动的终点。</p> <p>额外的功能：为了让转换看起来厉害一些，它还能创建 \t 变化。第二句不仅可以用于位置的调整，也可以拥有其它特效属性。任何不同于第一句的标签，会用于生成 \t 特效。</p> <p>你可以设置第二句的时间到最后一帧，脚本会只用第一句的开始/结束时间。同时第二句的文字也不会起作用。你可以设置第二句的开始时间在第一句之后，并勾选 keep both，这样在第一句转换成第二句之后，依然能维持第二句的时间。</p> <p>勾选 Rotation acceleration，可以确保旋转变走最小的路径，例如358-359-0-1-2，而非358-357-356-...-4-3-2。</p> <p>如果上下两句的 \pos 是相同的，那么只会进行 \t 样式的转化而非位置变化。你应当只选择一句，而非两句。</p> <p>times 可以让 \move 有时间戳。</p>
<p>Horizontal move means Y2 will be the same as Y1 so that the sign moves in a straight horizontal manner.</p> <p>Same principle for vertical.</p>	<p>Horizontal：把 \move 的前后Y坐标统一，使平移路径为一条水平直线。</p> <p>vertical 类似，为统一为竖直移动。</p>
<p>Multimove：When first line has \move and the others have \pos, \move for them is calculated from the first one.</p>	<p>Multimove：当第一句有 \move 且后面几句有 \pos，可以将第一句 \move 的位移复制到后几句。</p>
<p>Clip2Move *: Move is calculated from first 2 points of a vectorial clip. Works for \pos, or adjusts \move.</p> <p>If you make 4 points, the move is from the centre of line 1 (pt. 1+2) and line 3 (pt. 3+4).</p> <p>If you want exactly vertical or horizontal move, check only X or Y.</p>	<p>Clip2Move：将字幕按照 \clip 中的两点相对位置进行移动。适用于 \pos，也可以用来调整已经存在的 \move。</p> <p>如果你有四个点，那么移动会选择第1、2个点的中点到3、4两点的中点。</p>

原文	译文
<p>ShiftMove *: Like teleport, but only for the 2nd set of coordinates, ie. X2 and Y2. Uses input from Teleport.</p> <p>'current frame' sets -end- timecode to current frame</p>	<p>ShiftMove: 将字幕 \move 的第二对坐标 (终点坐标) 进行移动, 位移量来自 Teleport 的输入框。</p> <p>勾选 current frame, 使移动的结束时间变为当前帧。</p>
<p>ShiftStart: Similarly, this only shifts the initial \move coordinates.</p> <p>'current frame' sets -start- timecode to current frame</p>	<p>ShiftStart: 与 ShiftMove 相似, 移动 \move 的第一对坐标 (起始坐标)。</p> <p>勾选 current frame, 使移动的开始时间变为当前帧。</p>
<p>Reverse Move: Switches the coordinates, reversing the movement direction.</p>	<p>Reverse Move: 交换始末坐标, 使 \move 的移动方向相反。</p>
<p>Move to *: Teleport input sets target coordinates for \move for all selected lines.</p> <p>'current frame' sets end timecode to current frame</p>	<p>Move to: 将所有的选中字幕移动到 Teleport 区中填入的坐标。 current frame 将当前帧设置为移动的结束帧。</p>
<p>Move clip: Moves regular clip along with \move using \t\clip.</p>	<p>Move clip: 使用 \t\clip 来使矩形的 \clip 与主字幕一起同步移动。</p>
<p>Randommove</p>	<p>Randommove: 有二级弹窗输入移动信息, 按照设定值随机生成一个 \move 的效果。</p>
<p>Kill Times: nukes timecodes from \move / \t tags. (Default: move; checked "\t": transforms)</p> <p>Full Times: sets timecodes for \move / \t to the first and last frame.</p> <p>Set Times: sets timecodes for \move / \t based on Teleport input. X=start, Y=end.</p> <p>Negative values count from end. X=0 -> 0, Y=0 -> end frame. Last 100ms: X=-100,Y=0.</p>	<p>Kill Times: 删除 \move 和 \t 标签中的时间戳。</p> <p>Full Times: 设置 move 和 \t 标签中的时间戳为字幕开始和结束时间。</p> <p>Set Times: 设置 move 和 \t 标签中的时间戳为 Teleport 区的输入值, x 为开始时间, y 为结束时间。负数表示相差的时间。</p>
<p>Overlap: For Default-style lines that end after the next line starts, this creates \move to get out of the way.</p> <p>disPosition field: ms to end the move before next line begins (0 = move ends as next line appears.)</p> <p>Force field: duration of the \move (0 or lower defaults to 300. Both can just be left at 0.)</p>	<p>overlap: 只对于默认 Default 样式生效。如果AB两句字幕时间重合, 那么 Aegisub会自动在重合的时间把两句字幕上下叠放, 而此脚本可以将“叠放”的效果变成移动效果生成 \move。</p> <p>在 disPosition field 左上输入框中填入希望在下一句开始前结束移动的时间间隔, 以ms为单位。(0表示在下一句出现时结束移动)</p> <p>在 Force field 左中输入框中填入 \move 持续的时间。(0或更小数值则为默认300ms)</p>

原文	译文
<p>zig-zag : This will split a line into several that will move back and forth between given points A and B over the duration of the current line. Each move between A and B has the duration of the given interval. Shifting means that with each round, the point (A/B) moves by the given distance.</p> <p>If you start at top left (A) and top right (B) and set 0,10 for both shifts, the text will move left-right and go (10 pixels) lower with each round, ending up at the bottom of the screen if given enough time. You can choose starting position, but with shifting, probably only A will work well.</p> <p>With 'middle', the second point is always A, and the move, being half the distance, will only last half the given interval, so the speed is the same.</p> <p>'Line's position' means Pos A will be the current position of the line.</p> <p>'Relative to A', if set for example to '0,100', means Pos B will be 100 pixels below Pos A.</p> <p>Using both these options at the same time allows for several selected lines to move in unison. If A is line's position and B is absolute, all lines will meet at point B and then go back to where they started.</p>	<p>zig-zag : 这个脚本会将一句字幕分开, 在当前行的持续时间内按照给定的时间间隔在给定的点A和B之间移动。 Shifting 意味着每一轮AB两点都按照给定距离移动。</p> <p>如果你开始在左上 (A) 和右上 (B) 并设置轮换 shift 为 0,10 , 文字会左右移动并且每一轮的位置向下降低10像素。你可以设置开始的位置, 但是通过移动, 可能只有A保持有效。</p> <p>勾选 middle , 第二点将永远是A, 而移动的效果, 将会只有一半的距离, 保持一半的时间间隔, 以及一般的速度。</p> <p>line position 指位置A将会是当前字幕的 \pos 位置。</p> <p>Relative to A , 即默认设置成 0,100 , 即位置B在位置A向下100像素。同时选中这些选项, 可以对选中的多句话进行同步移动。如果A位置勾选为了字幕的位置而B是绝对位置, 所有字幕将会在B相遇并回到起始位置。</p> <p>译者注: 目前并没有成功模拟实现这个功能, 怀疑bug。</p>
<p>* times will add timecodes to \move for these functions (from Teleport if given, or just first/last frames).</p>	<p>* Times 将会对这些功能生成的 \move 标签加入时间戳 (如果 Teleport 区有设置则为 Teleport 的时间, 如果没有则默认字幕的始末时间。)</p>

Morphing Ground 变形区



原文	译文
<p>Round Numbers: rounds coordinates for pos, move, org and clip depending on the 'Round' submenu.</p>	<p>Round Numbers: 根据下方选项来四舍五入 \pos \move \org \clip 的坐标值到整数。</p>
<p>line2fbf: Splits a line frame by frame, ie. makes a line for each frame. If there's \move, it calculates \pos tags for each line. If there are transforms, it calculates values for each line. It should deal with all transforms, including inline tags and acceleration. Move and transforms can have timecodes. (Though some calculations may end up about 1% off.) \fad is supported too, but may not be entirely accurate with complex alphas. Very complex lines with multiple transforms are likely to break in some way.</p> <p>clip2fbf - clips will be shifted along with \move. (Don't use with clip transforms.)</p>	<p>line2fbf: 把一句字幕改为 Frame by frame 逐帧字幕。如果句中有 \move, 则会为每一句计算 \pos 位置。如果有 \t, 也会每一句单独计算标签。适用于所有 \t 变化, 包括标签和加速度。 \move 和 \t 都支持时间戳。(某些结果可能有1%的不精准)。同样支持 \fad, 但对于复杂的透明效果可能支持不精准。非常复杂的、有多重 \t 的句子可能不精准。如果勾选了 clip2fbf, 那么 \clip 遮罩将会随着 \move 一起移动。(译者注: 类似 Move clip。)</p>
<p>Joinfbflines: Select frame-by-frame lines, input number X into Force Field, and each X lines will be joined. (same way as with 'Join (keep first)' from the right-click menu)</p>	<p>Joinfbflines: 选中几条 fbf逐帧字幕, 向 Force Field 输入框填入数值 X, 使每 X 条字幕合并在一起。(功能类似右键的 合并 选项, 并没有计算, 只是单纯合并。)</p>
<p>Set origin: sets \org based on Teleport input. With 'Warp', it's set by given distance from \pos.</p>	<p>Set Origin: 根据 Teleport 区填入的值设置 \org。如果勾选 warp, 则根据给定的和 \pos 的距离来设置。</p>
<p>Move with clip: Moves vectorial clip on fbf lines based on \pos tags.</p> <p>Note: For decimals on v-clip coordinates: xy-vsfilter OK; libass rounds them; regular vsfilter fails completely.</p>	<p>Move with clip: 移动矢量的 \clip 基于 \pos 标签。</p> <p>译者注: 这是一个很实用的功能, 是高级版的 move clip, 可以适用于矢量 \clip 的移动。具体步骤:</p> <ol style="list-style-type: none"> 1. 创建一个带有 \pos 标签变化的 fbf 逐帧 移动字幕。 2. 每一句字幕都应有相同的 \clip 标签。 3. 应用脚本, 将自动以第一行为基准, 根据 \pos 的相对位置变化来变动后续 \clip 的位置。
<p>Transform clip</p>	<p>Transform clip: 一个方便调整矩形 \clip 移动的二级选框。</p>

原文	译文
<p>Calculate origin: This calculates \org from a tetragonal vectorial clip you draw.</p> <p>Draw a vectorial clip with 4 points, aligned to a surface you need to put your sign on.</p> <p>The script will calculate the vanishing points for X and Y and give you \org.</p> <p>Make the clip as large as you can, since on a smaller one any inaccuracies will be more obvious.</p> <p>If you draw it well enough, the accuracy of the \org point should be pretty decent.</p> <p>(It won't work when both points on one side are lower than both points on the other side.)</p> <p>See this blog post for more details: http://unanimated.hostfree.pw/itw/tsblok.htm#origin</p>	<p>Calculate Origin: 根据绘制的四边形 \clip 计算 \org。</p> <p>绘制一个拥有四点的矢量 \clip，对齐到需要显示的平面。脚本将会计算出 x 和 y 的消失点，并给出 \org。尽量画大，因为小的四边形误差比较大。（在同一边的两个点都比另一边的低时不生效）</p> <p>更多介绍: http://unanimated.hostfree.pw/itw/tsblok.htm#origin</p> <p>译者注：实际上并没有打 \fax 方便。</p>
<p>Set Rotation: adds selected rotation tags with the value from the 'rotation' menu. (You can get multiples of 30 using the Aegisub tool while holding Ctrl.)</p>	<p>Set Rotation: 根据选中的下方 Rotation 旋转选项卡来添加 \frz 标签。（按住Ctrl键的同时，可以使用Aegisub工具获得30的倍数。译者注：不知道是什么意思）</p>
<p>Rotate 180: rotates text by 180 degrees from current values of selected rotations (frx, fry, frz).</p>	<p>Rotate 180: 根据选中来旋转 \frx \fry \frz 180度。</p>
<p>Negative Rot: keeps the same rotation, but changes to negative (350 -> -10), which helps with transforms.</p>	<p>Negative Rot: 处理角度用。保持相同的角度，但是标识角为相反符号。例如：350 -> -10。在 \t 变化时会有用。</p>
<p>Spin Doctor: These are settings for the two Spin macros. The macros change tag values based on this.</p> <p>These are settings for the Spin macros. Choose a tag you want to affect with them and step. Default (without loading this GUI) is frz & 1.</p>	<p>Spin Doctor: 不会用。</p>
<p>Vector2Rect / Rect.2Vector: converts between rectangular and vectorial clips.</p>	<p>Vector2Rect / Rect.2Vector: 在矩形和矢量 \clip 间转换。</p>
<p>Clip scale: Use Force field to set the X factor in "clip(X,m ", and the clip will be recalculated accordingly.</p>	<p>Clip scale: 根据 Force field 填入的倍数，重新计算缩放矩形。</p>
<p>Clip2Scale fbf: Adjusts \fscx and \fscy fbf based on width and height of a 2-point clip. (Check online manual.)</p>	<p>Clip2Scale fbf: 根据两点的向量，调整 \fscx 和 \fscy。</p>

原文	译文
<p>Clip Info: For signs with different size on each end, draw a 4-point vectorial clip that reflects the sizes. The point is mainly to get the difference in height between left and right sides of a tetragon. If the right size is twice the height of left, 'fsc' will put \fscx200\fscy200 before last character. More generally, you can get measurements for any two lines and the ratio between them. Line A is clip points 1-2, line B is 3-4. This can be used for calculating the scaling of various things. Applies only to active line.</p>	<p>clip info: 对于两端大小不同的标签，画一个反映大小的4点矢量剪\clip。这一点主要是为了得到四边形左右两侧的高度差。如果右侧大小是左侧高度的两倍，则“fsc”会将\fscx200\fscy200 放在最后一个字符之前。更一般地说，您可以获得任意两条线的测量值以及它们之间的比率。线A是剪辑点1-2，线B是3-4。它可用于计算各种事物的比例。适用于活动（非注释）的字幕行。</p>
<p>[Un] hide clip: Hides an existing clip in a comment / returns it back. (This has its own macro for hotkeying.)</p>	<p>[Un] Hide clip: 隐藏/显示\clip。有独立的无GUI选项，可以配置相应的热键。</p>
<p>Find Centre: A useless function that sets \pos in the centre of a rectangular clip.</p>	<p>Find Center: 一个没啥用的功能，让\pos处于矩形\clip的中心。</p>
<p>Randomise: randomises values of given tags. With \fs50 and value 4 you can get fs 46-54. For regular type tags, you can input multiple ones with commas between them. Also works for alpha, but not colours. Negative values only happen where applicable. Available for \t times.</p>	<p>Randomise: 随机化给定的标签。例如对\fs50执行随机4，将会得到\fs45-54。对于普通的标签，可以用逗号分隔，进行不同区间的随机化。对透明度也生效，但对颜色不可以。只有在适用时才可以用负值。对\t有效。</p>
<p>Letterbreak: creates vertical text by putting a linebreak after each letter. wordbreak: replaces spaces with linebreaks.</p>	<p>Letter break: 在原句每个字母间加入\n换行符。 word break: 在原句每个单词间加入\n换行符。</p>
<p>Extend Mask: Use Teleport X and Y fields to extend a mask in either or both directions. This is mainly intended to easily convert something like a rounded square to another rounded rectangle. Works optimally with a 0,0 coordinate in the centre. May do weird things with curves. When all coordinates are to one side from 0,0, then this works like shifting.</p>	<p>Extend Mask: 根据 Teleport 区填入的值设置来扩大蒙版。这主要是为了方便地将一个圆形的正方形转换成另一个圆形的矩形。以0,0为中心的坐标为最佳工作状态。对于曲线可能会做一些奇怪的事情。当所有的坐标都在0,0的一边，那么这就像移位一样。 译者注：不知道是干什么的</p>

原文	译文
<p>Flip mask: Flips a mask so that when used with its non-flipped counterpart, they create hollow space. For example you have a rounded square. Duplicate it, extend one by 10 pixels in each direction, flip it, and then merge them. You'll get a 10 px outline.</p>	<p>Flip Mask：翻转遮罩，当与非翻转的遮罩一起使用时，它们会产生空的空间。</p> <p>例如，你有一个圆形的正方形，将它复制，在每个方向上延伸10个像素，翻转它，然后将它们合并。复制它，向每个方向延伸10个像素，翻转它，然后合并它们。你会得到一个10 px的轮廓。</p> <p>译者注：不知道是干什么的</p>
<p>Adjust Drawing: (You must not have an unrelated clip in the line.)</p> <ol style="list-style-type: none"> 1. Creates a clip that copies the drawing. 2. You adjust points with clip tool. 3. Applies new coordinates to the drawing. 	<p>Adjust Drawing：调整绘图（你不能在行中有一个不相关的蒙版。）</p> <ol style="list-style-type: none"> 1. 创建一个复制图形的蒙版 2. 你可以用clip工具调整蒙版的控制点 3. 向绘图应用新的坐标点。 <p>译者注：不知道是干什么的</p>
<p>Randomask: Moves points in a drawing, each in a random direction, by a factor taken from the Force field.</p>	<p>Randomask：随机化蒙版。</p> <p>在 Force 区填入需要随机的力度，然后可以随机移动绘图的坐标点。</p>

Cloning Sequence 克隆区

此工具可以克隆一个序列的字幕。

原文	译文																						
<p>This copies specified tags from first line to the others.</p> <p>Options are position, move, origin point, clip, and rotations.</p> <p>replicate missing tags: creates tags if they're not present</p> <p>stack clips: allows stacking of 1 normal and 1 vector clip in one line</p> <p>match type: if current clip/iclip doesn't match the first line, it will be switched to match</p> <p>combine vectors: if the first line has a vector clip, then for all other lines with vector clips the vectors will be combined into 1 clip</p> <p>copy rotations: copies all rotations</p> <p>With Teleport input, you can shift X and Y coordinates more each line by the given values.</p>	<p>这个功能可以从第一句复制特效标签到其他句。</p> <p>标签包括位置、移动、原点设置、蒙版和旋转。</p> <p>会自动复制缺失的标签：如果不存在，则自动创建。</p> <p>堆叠剪辑：允许一句普通字幕叠加一个 clip 蒙版。</p> <p>匹配类型：如果当前的蒙版 clip 或反蒙版 iclip 与第一行不匹配，则将切换为匹配</p> <p>合并矢量。如果第一行有一个矢量剪辑，那么对于所有其他有矢量剪辑的行，矢量将被合并成一个剪辑。</p> <p>复制旋转：复制所有的旋转</p> <p>使用 Teleport 取输入的数值，你可以设置每一次变化的步长。例如：</p> <table><tr><td>ult</td><td>{\pos(0,0)}A</td></tr><tr><td>ult</td><td>B</td></tr><tr><td>ult</td><td>C</td></tr><tr><td>ult</td><td>D</td></tr><tr><td>ult</td><td>E</td></tr></table> <p>如图，设置 Teleport 区为(10,30)，则克隆后会变为</p> <table><tr><td>样式</td><td>文本</td></tr><tr><td>Default</td><td>{\pos(0,0)}A</td></tr><tr><td>Default</td><td>{\pos(10,30)}B</td></tr><tr><td>Default</td><td>{\pos(20,60)}C</td></tr><tr><td>Default</td><td>{\pos(30,90)}D</td></tr><tr><td>Default</td><td>{\pos(40,120)}E</td></tr></table>	ult	{\pos(0,0)}A	ult	B	ult	C	ult	D	ult	E	样式	文本	Default	{\pos(0,0)}A	Default	{\pos(10,30)}B	Default	{\pos(20,60)}C	Default	{\pos(30,90)}D	Default	{\pos(40,120)}E
ult	{\pos(0,0)}A																						
ult	B																						
ult	C																						
ult	D																						
ult	E																						
样式	文本																						
Default	{\pos(0,0)}A																						
Default	{\pos(10,30)}B																						
Default	{\pos(20,60)}C																						
Default	{\pos(30,90)}D																						
Default	{\pos(40,120)}E																						

Teleportation 坐标移动区

一个非常实用而强大的工具，可以使用 Teleport 指定单句中所有坐标的位移。

原文	译文
<p>Teleport shifts coordinates for selected tags (\pos\move\org\clip) by given X and Y values. It's a simple but powerful tool that allows you to move whole gradients, mocha-tracked signs, etc. Note that the Teleport fields are also used for some other functions, like Shiftstart and Shiftmove. These functions don't use the 'Teleportation' button but the one for whatever part of HR they belong to. 'mod' allows you to add an extra factor applied line by line.</p> <p>For example if you set '5' for 'X', things will shift by extra 5 pixels for each new line.</p> <p>c1 and c2 control whether Teleportation affects top left and/or bottom right corner of a rectangular clip. This means that for both X and Y, you can move either one or both sides of the clip.</p> <p>'exp' means a rectangular clip won't be moved but expanded. X 10 means the clip will expand by 10 to the left and right. This ignores the c1/c2 settings.</p> <p>Warped Teleport adds an additional value for each new line.</p>	<p>Teleport 利用给定的(x,y)变化量, 移动选定标签 \pos\move\org\clip 的坐标。</p> <p>这是一个简单而强大的工具, 移动所有的渐变、Mocha追踪等。</p> <p>注意 Teleport 输入框同样可以用于其他的功能, 例如 Shiftstart 和 Shiftmove。</p> <p>这些功能不使用 Teleportation 按钮, 但是使用这个区域的数值。</p> <p>mod 允许你在生成的逐句字幕中添加额外的变量。</p> <p>例如, 你在 x 区设置了 5, 那么所有新的行中内容会额外移动5像素。</p> <p>c1 和 c2 控制了是否调整矩形 clip 的左上角和/或右下角。这意味着对于 x 和 y, 你可以移动 clip 一边或者两边。</p> <p>exp 意味着矩形蒙版不会被移动而被扩大。x10 指左右扩大十倍。这个选项会忽略前文提到的 c1、c2。</p> <p>Warped Teleport 给每一行添加一个额外的值。</p>