运动对象视频修改流程手册

1. 计算过程部分

电子表格[1px=Yours]中 根据计算可以得到实际屏幕尺寸下 0.1 m/s 的基准视频的占用时间(s)，来制作基本视频素材。



(上图为假设参数，下文依照此图计算数据)

特别注意：

控制时间的要素是：

1. 屏幕宽度像素值(px)、实际尺寸(m)
2. 视频[文字运动]中起始位置(px)与终止位置(px)

现在无关化处理车牌对象的大小，如果需要调整大小就在[文字运动]序列中调整即可

计算速度公式： V = S / t

我们通过固定距离，改变时间从而改变速度。因此距离的确定十分关键。

像素是矩阵概念，矩阵元素与元素间没有距离概念，因此确定实际长度，需要联系实际播放设备尺寸以及实际播放设备的像素，进而求得比例尺R(mm/px)。

举例：

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 第一行示意：24个像素 | | | | | | | | | | | | | | | | | | | | | | | |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 本行示意：实际距离 S (mm) | | | | | | | | | | | | | | | | | | | | | | | |

因此比例尺R = S (mm) / 24 (px)

计算过程为：

1. 得到 展示设备实际尺寸后 比例尺(m/px)=实际尺寸/px
2. 得到 比例尺(mm/px)=比例尺(m/px)\*1000
3. 得到 屏幕运动距离(mm)=(终止位置(px)-起始位置(px))\*比例尺(mm/px)
4. 得到 屏幕运动距离(m)= 屏幕运动距离(mm)/1000
5. 根据要求速度计算得到视频片段时间(s) = 屏幕运动距离(m)/要求速度(m/s)
6. 得到基准时间

Ps：基准时间就是最低速率下的片段的时间，也是最长时间；通过变化这个片段就可以得到任何速度片段。

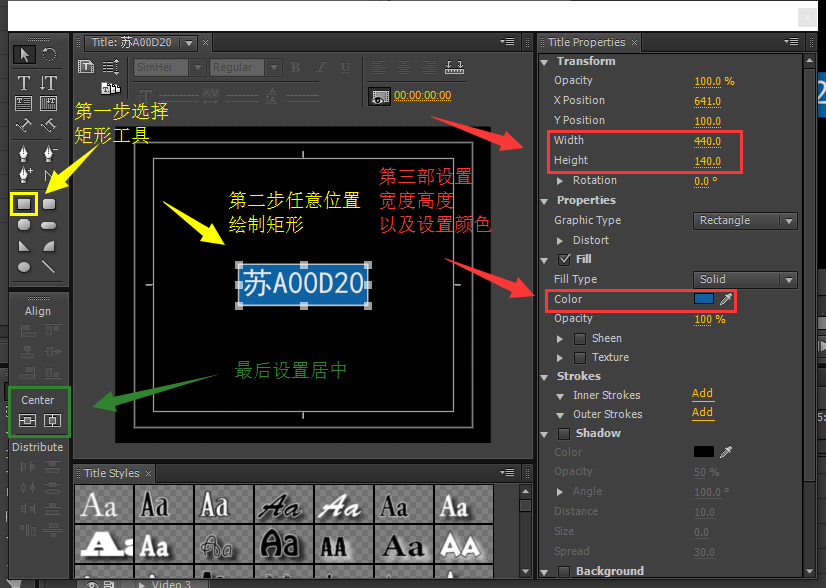
1. 视频制作部分 [2017.04.20方案]

我们视频采用嵌套模式，(所谓嵌套就是将重复的特效让他在序列层面完成，例如将多个车牌对象Title的速度设置转移到整个序列的调整部分)。

我们的视频素材层级关系为：

解释：(项目序列以1280\*1080 30帧格式为例，之后讲解新建序列)

1. 车牌使用Title制作而成，包括背景矩形(宽度、高度)与文字(黑体、字体大小)
2. Title中背景矩形设置：

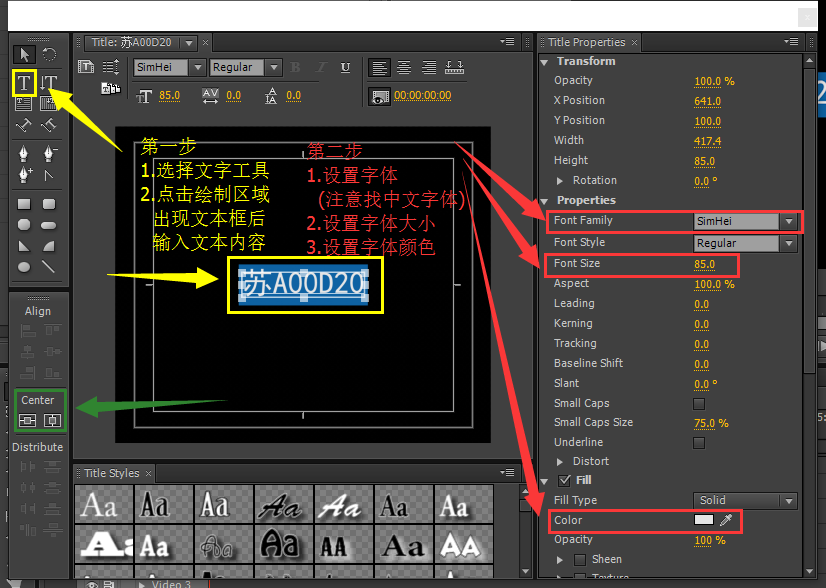


1:矩形大小修改: Width 与 Height 的值；

2:矩形颜色修改: Color 处修改

3:从Center处，让矩形居中显示

1. Title中文字设置：



注意字体选择，一定要让中文显示正确！推荐使用SimHei、Adobe Heiti Std字体；字体颜色一般默认即可；字体大小调整差不多后，居中处理就完成了。

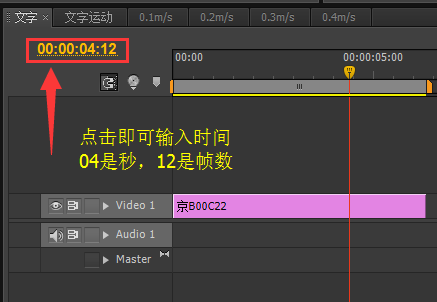
1. 文字序列有40个轨道，用于存放40个不同内容的车牌Title，需要设置时间长度，不需要设置动画：

Step 1:首先将之前做的车牌Title拖入[文字]序列中：

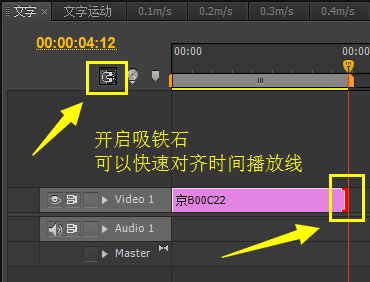
|  |  |
| --- | --- |
| C:\Users\Anshare_LY\AppData\Roaming\Tencent\Users\792493310\QQ\WinTemp\RichOle\HQRTC]$WVT)W5AW]F5@PJ%M.png | C:\Users\Anshare_LY\AppData\Roaming\Tencent\Users\792493310\QQ\WinTemp\RichOle\3J6}K9MQ`VDFNABO6K6OI~0.png |
| 选择素材 | 拖入轨道 |

Step 2:然后根据表格计算的时间，调整素材长度，并截取：

我们得到0.1m/s的时间是4.39 s ，因为我们是30帧每秒，所以约等于4秒12帧；



Step 3:然后用鼠标选择车牌Title尾部，拖拽至时间播放线处(4秒12帧处)：



Step 4:之后增加轨道数量：

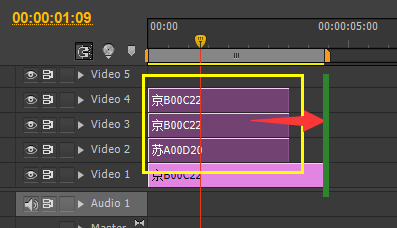
在轨道左侧的轨道名称部分点击右键选择增加轨道(Add Tracks…)

|  |  |
| --- | --- |
| C:\Users\Anshare_LY\AppData\Roaming\Tencent\Users\792493310\QQ\WinTemp\RichOle\}~5C4K`TKS0QXS$1)DX~Y[8.png | C:\Users\Anshare_LY\AppData\Roaming\Tencent\Users\792493310\QQ\WinTemp\RichOle\V}SZLW([3}P2)[M0}WLAZ45.png |
| 在此区域右键选择增加轨道 | 设置增加轨道的页面 |

Step 5:之后在新建的轨道上增加Title文件即可(Title文件复制之前做的Title文件，然后修改内容就可以快速的生成新的Title文件了)。之后依旧需要调整到(结束位置：4秒12帧处)

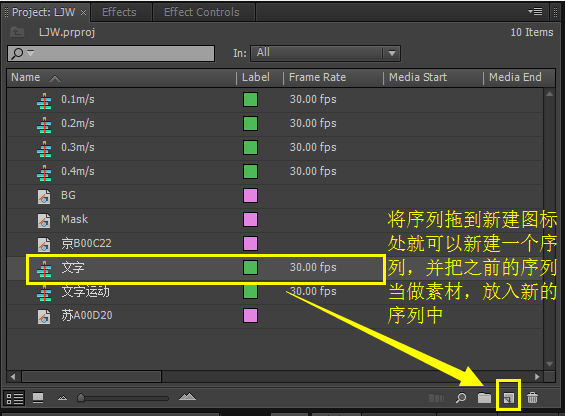
技巧：

1. 先复制出来39个新的Title，然后分别修改内容；
2. 然后将39个新的Title文件分别放入视频轨道中，一个轨道一个；
3. 使用快捷键 **[Ctrl+A]** 全选，然后按住**Shift**点击编辑好时间长度的第一轨道素材，完成排除第一轨道素材的其他新加入素材的选择；



1. 整体拖到第一轨道素材的结束处完成批量素材时间修改；
2. 按住**Shift**点击轨道左侧的小眼睛处就可以实现**全部开启**或者**全部关闭。**
3. 文字运动序列包括文字序列以及Mask(用于参照确定运动起止位置用，渲染时关闭轨道的眼睛)

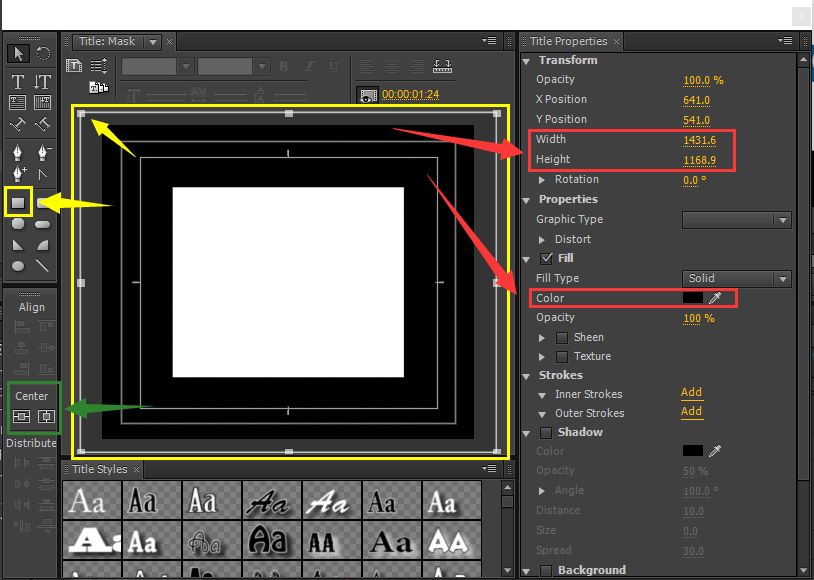
Step 1:首先新建[文字运动]序列：



提醒：新建完序列后，序列名称与之前名称一致，所以需要修改新生成序列的名称，变为[文字运动]序列。

Step 2:然后制作Mask：

1. 制作大的黑色背景矩形：

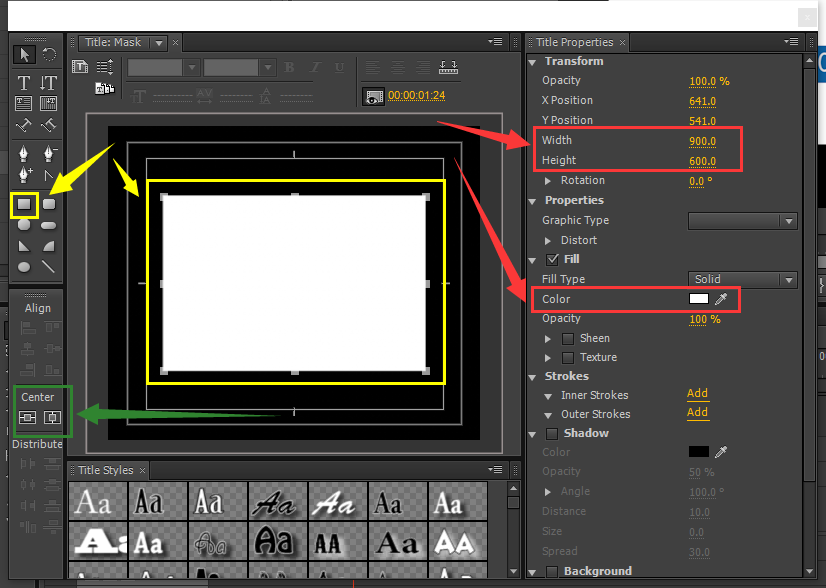


本步骤制作的矩形与之前制作背景矩形步骤相同，但是要注意的是，黑色背景矩形尺寸一定要大过序列尺寸(尺寸>1280\*1080)，而且颜色一定是纯黑色！

关于遮罩原理：

黑色像素值为0，白色像素值为1。用遮罩乘以其他图像，则只有白色的部分可以显示，其余黑色的部分则不显示。因此一定要纯黑色与纯白色！

1. 制作小的可见部分矩形：



本步骤制作的矩形与之前制作背景矩形步骤相同，但是要注意的是，根据遮罩大小设置矩形，例如(900\*600)，而且颜色一定是纯白色，最好要记得居中处理。

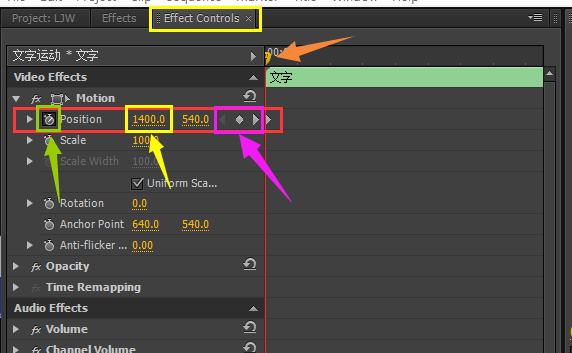
Step 3:为[文字]序列制作运动效果：

1. 将[文字]序列拖入第二轨道，然后将Mask放入第一轨道，为文字提供背景参考：

|  |  |
| --- | --- |
| C:\Users\Anshare_LY\AppData\Roaming\Tencent\Users\792493310\QQ\WinTemp\RichOle\KXMXB9MRYHTFZB2DDT$(LHO.png | C:\Users\Anshare_LY\AppData\Roaming\Tencent\Users\792493310\QQ\WinTemp\RichOle\$Q2WNCT9@NPH1_478M~3%AS.png |
| 轨道设置 | 显示效果 |

1. 设置[文字]序列的起止位置Position关键帧：

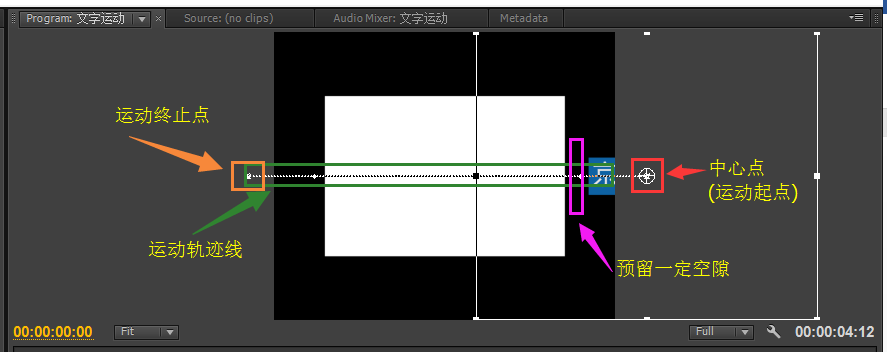
选中[文字]序列，然后在效果控制中修改Position属性：



提示：

1. 绿色框中的小闹钟(计时器)开启关键帧记录功能，不打开不会记录关键帧；
2. 黄色框部分是X轴的坐标(px)，打开小闹钟后只要变动X轴坐标便会记录关键帧；
3. 粉色框中是关键帧的便捷操作，向左箭头功能为迅速将播放时间线打到上一个关键帧处，向右箭头表示迅速将播放时间线打到下一个关键帧处。中间的点功能为添加关键帧；

我们从最开始(0秒0帧处)的地方打一个关键帧，然后调整X轴坐标使得[文字]序列完全不在Mask白色部分之上；然后我们找到终止时间(4秒12帧)，同样修改X轴坐标使得[文字]序列完全不在Mask白色部分之上，于是当点击Motion时得到如下图：

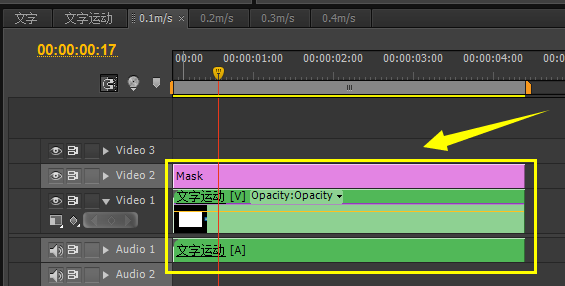


[文字运动]序列到此制作完毕！

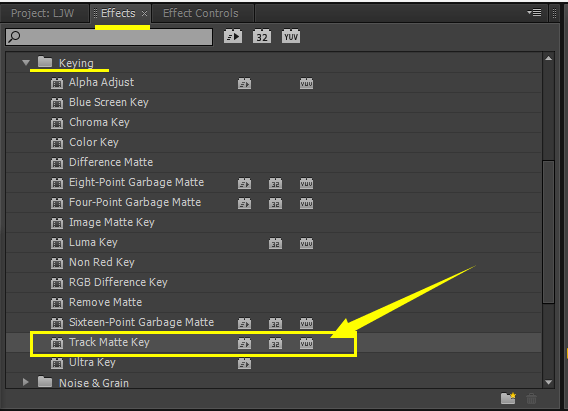
1. 制作0.1m/s序列，以及其他速度序列：

Step 1:制作0.1m/s序列：

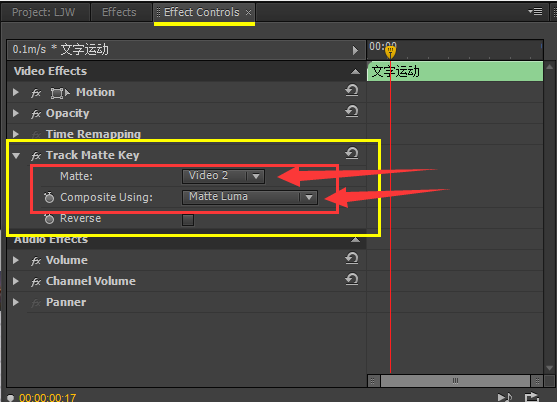
同样的，将[文字运动]序列拖入新建图标，生成一个新的序列，并重命名为[0.1m/s]序列；然后在第二轨道中拖入Mask素材，并调整素材长度与[文字运动]序列一致：



在特效中找到：Video Effect -> Keying -> Track Matte Key 特效(轨道遮罩特效)



将特效拖入[文字运动]素材中：



提示：

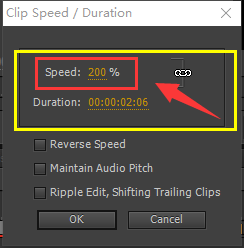
1. Matte表示选择要当遮罩层的轨道，目前我们将Mask素材放到了 Video 2轨道上了，所以选择“Video 2”
2. Composite Using 表示遮罩方法，我们选择Matter Luma(亮度遮罩)，也就是白色的地方保留，黑色的地方不保留

于是我们就得到了这样给的效果：

|  |  |
| --- | --- |
| C:\Users\Anshare_LY\AppData\Roaming\Tencent\Users\792493310\QQ\WinTemp\RichOle\%EX34@9M3J6V5H14W{XX`1D.png | C:\Users\Anshare_LY\AppData\Roaming\Tencent\Users\792493310\QQ\WinTemp\RichOle\SB}~1GGRXXN1]{8)RV1A2CL.png |
| 使用遮罩效果前 | 使用遮罩效果后 |

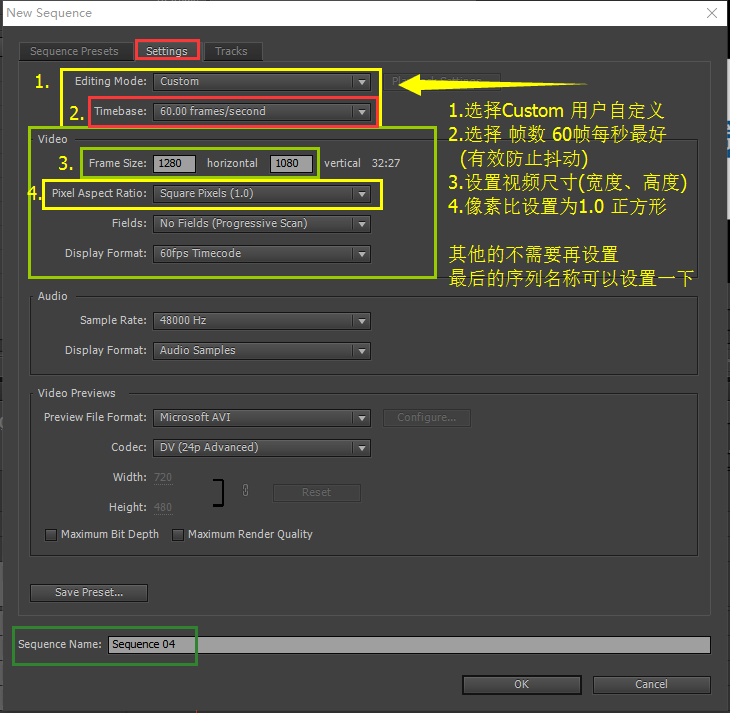
Step 2:制作其他速度序列：

同样的，将[0.1m/s]序列拖入新建图标，生成一个新的序列，并重命名为[0.2m/s]序列(举例，其他的以此类推)，然后右键点击轨道上的[0.1m/s]素材，选择“Speed/Duration…”：



提示：调整速度百分比就可以获取到其他速度了。100%是原速度，200%是2倍速度(0.2m/s)，300%是三倍速度(0.3m/s)…… 以此类推。

1. 新建序列设置

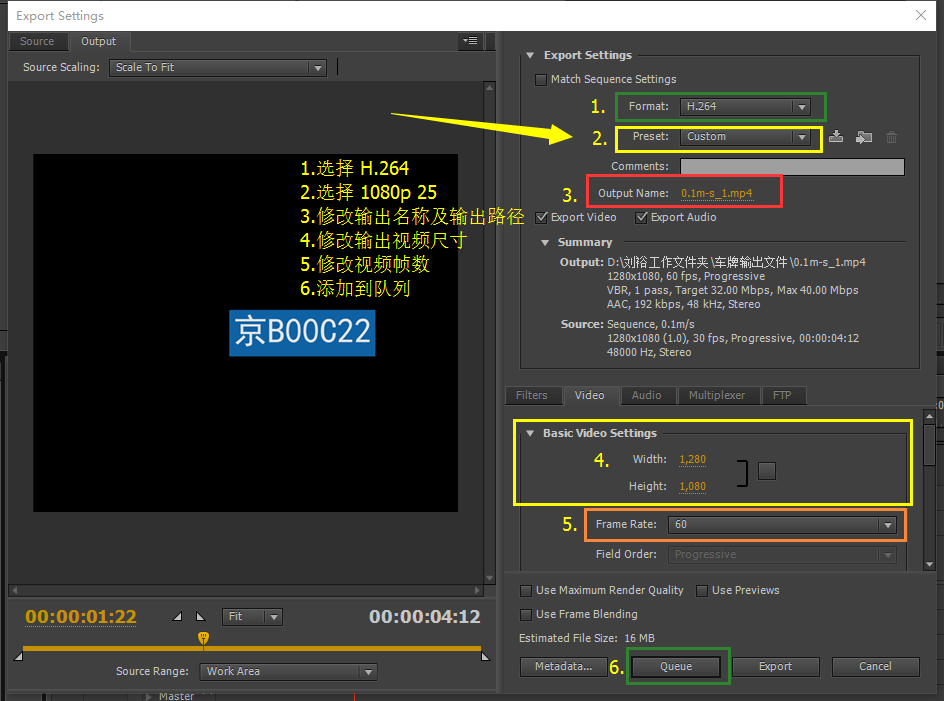


第一页是序列格式预设，没有我们需要的格式，所以我们直接选择 “Setting”进入详细设置页面。

为了满足实验要求，我们自定义视频格式。

1. 渲染设置

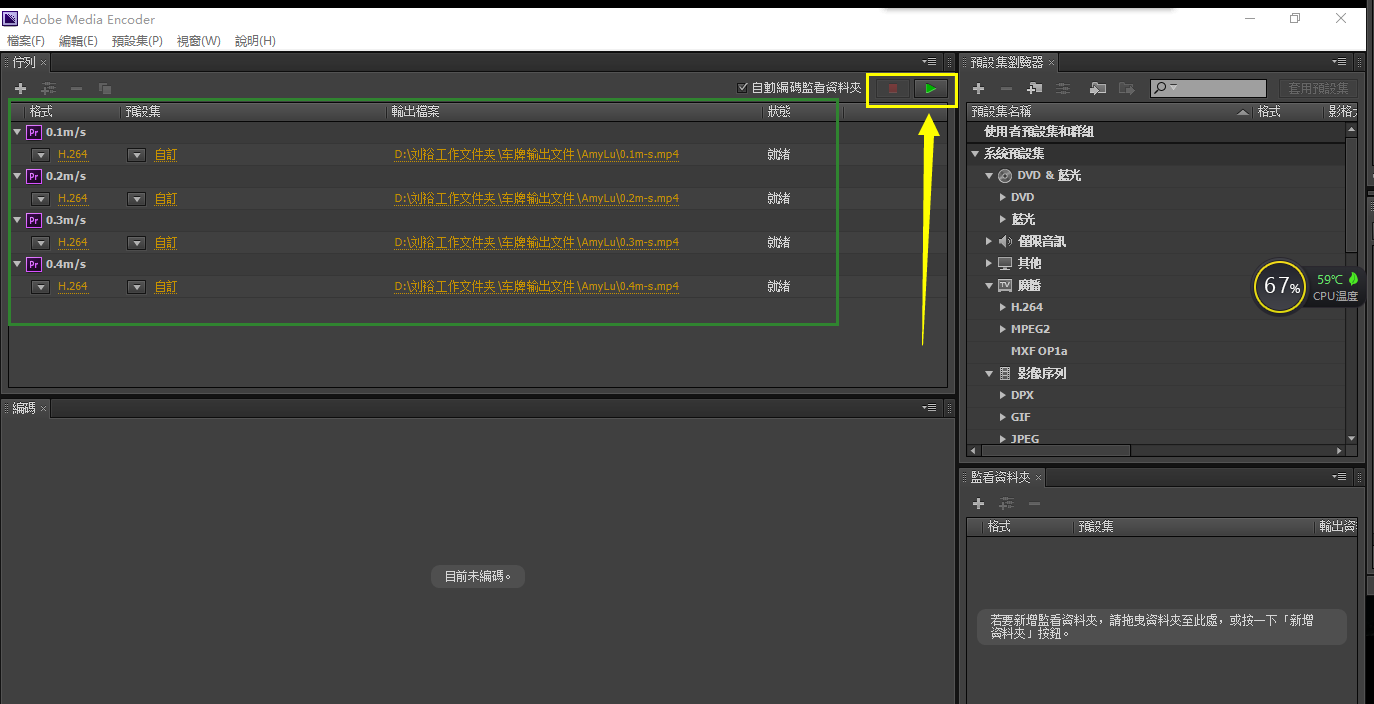
经过测试 H.264 1080 P 25 利用这个修改得到的渲染模式效果最好



提示：

1. Queue是队列整体输出，用呆了 Adobe Media Encoder 软件，可以完成批量输出的功能
2. Export是直接输出，即单个文件输出
3. 输出队列

当使用Queue队列输出后，就会弹出Adobe Media Encoder软件：



点击右上角的绿色箭头按钮就可以开始批量视频渲染导出了。