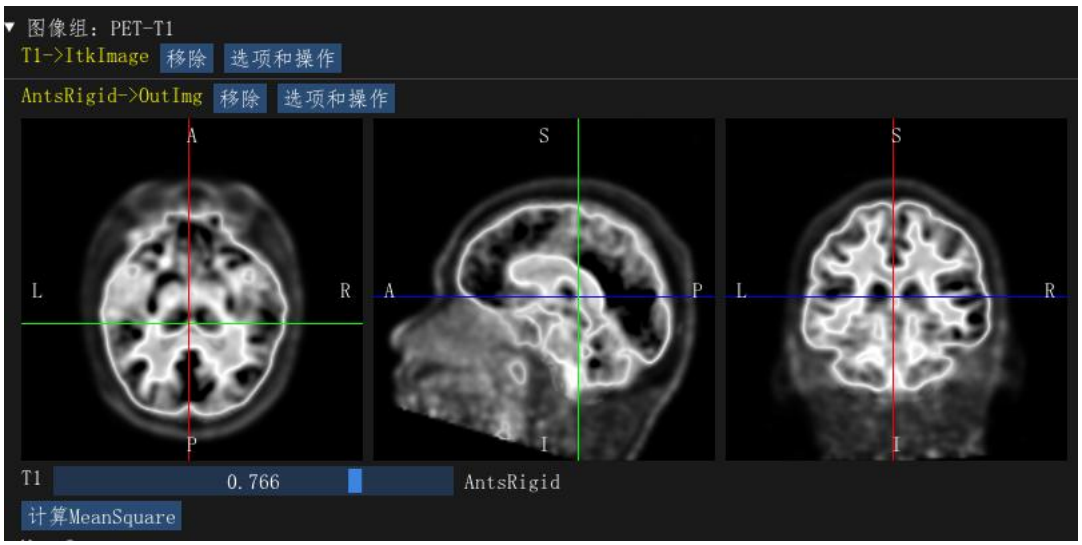
渲影医析 Lab 实战案例（2）结构 MRI PET FMRI 同步配准（Co-Register）

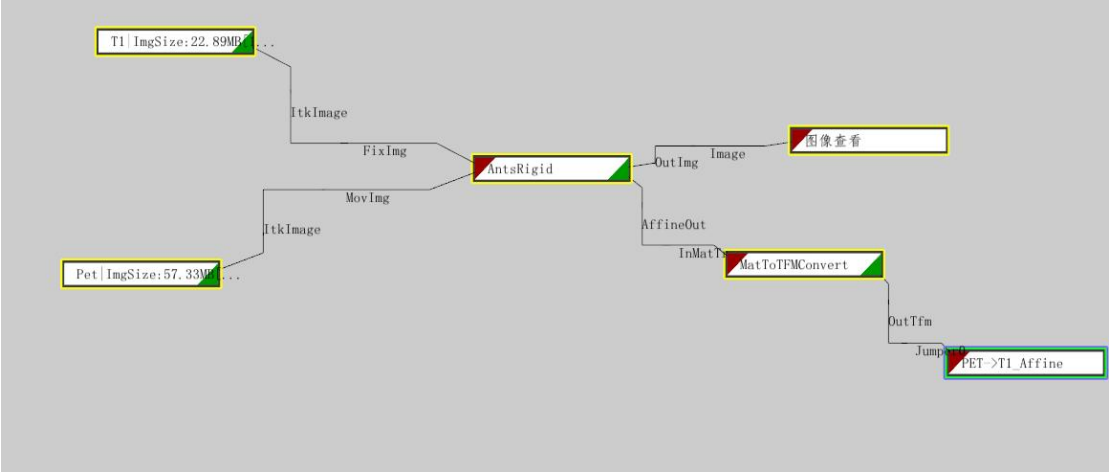
T1-PET Co-Register 到 MNI 模板

T1-PET 同步配准指的是，先将 T1 配准到 MNI 模板再将同病人的 PET 匹配到 MNI 的过程。这里解决的主要问题是 PET 往往无法直接配准到 MNI 结构核磁专用模板，而利用同病人 PET-结构核磁之间的刚体配准结果可以间接达到此目的。该原理可以同样用于 T1、FMRI、PET 等多种模态之间的配准。从根本上理解这个过程，用户就可以自行设计各种其他模态数据的处理流程。

步骤 1：基于实战案例（1）中介绍的同病人 PET->T1 刚体配准



基于实战案例（1）中介绍的同病人 PET->T1 刚体配准结果，如上图。

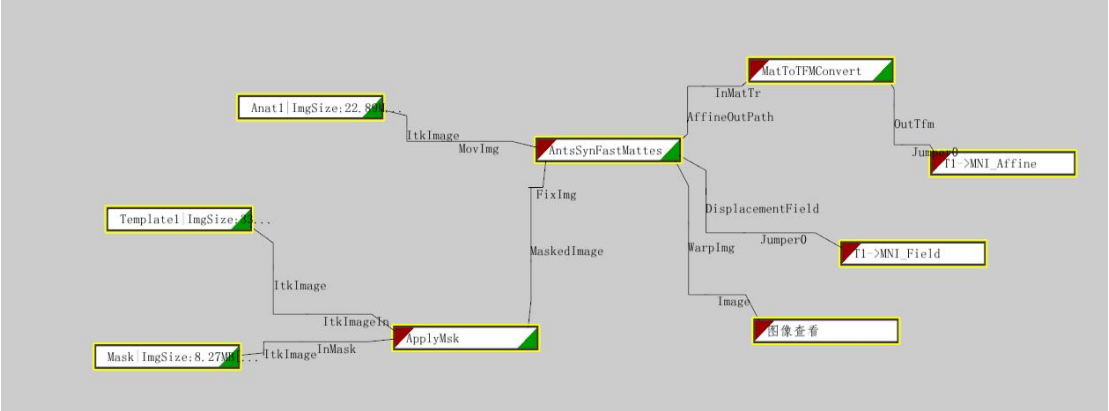


*步骤 1 中生成的配准后图像并不会直接使用，我们使用 **Ants Rigid** 生成的 **Affine** 矩阵，用一个跳跃器收集这个矩阵输出，并重新命名为 **PET->T1_Affine**。*

步骤 2 ： T1 和 MNI 模板配准。



*基于实战案例（1）中介绍的同病人 **T1->MNI** 模板配准结果，如上图。*



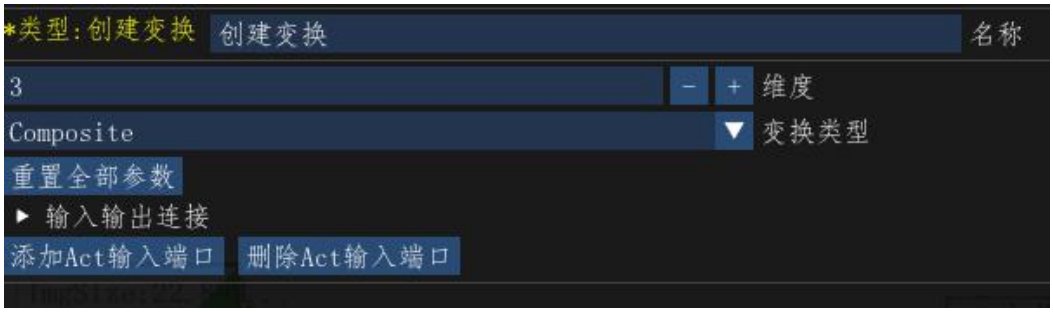
除了 Wrap 到 MNI 的 T1 图像以外，这个步骤还需要两个额外的输出：T1->MNI_Affine 仿射配准矩阵和 T1->MNI_Field T1 到 MNI 的形变场。

步骤 3：将所有变换一次性应用到 PET 上。

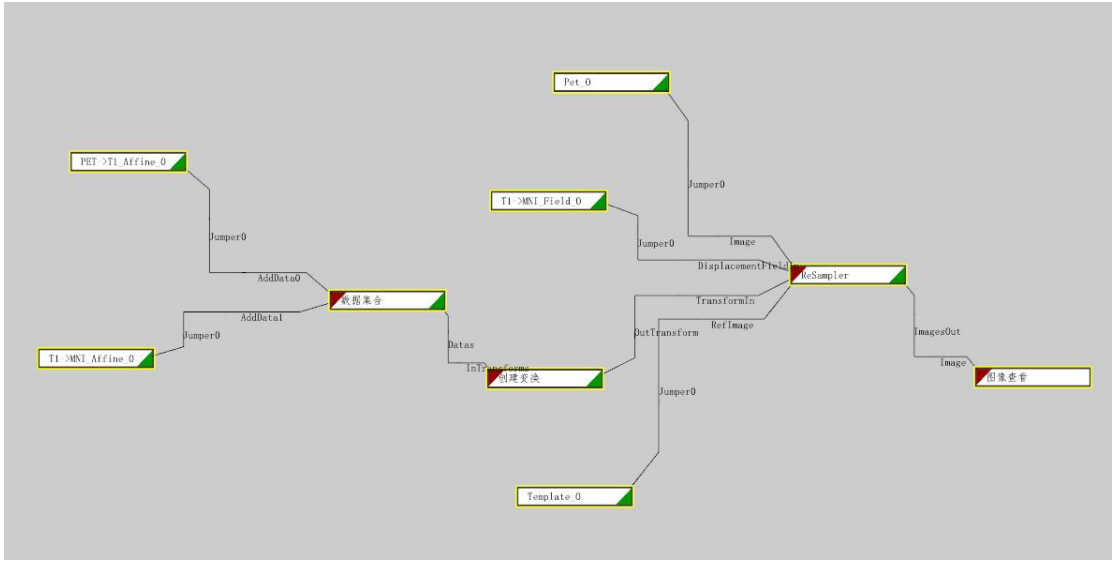
首先创建一个复合变换：



医学影像处理->基本操作->创建变换



变换类型选择 Composite

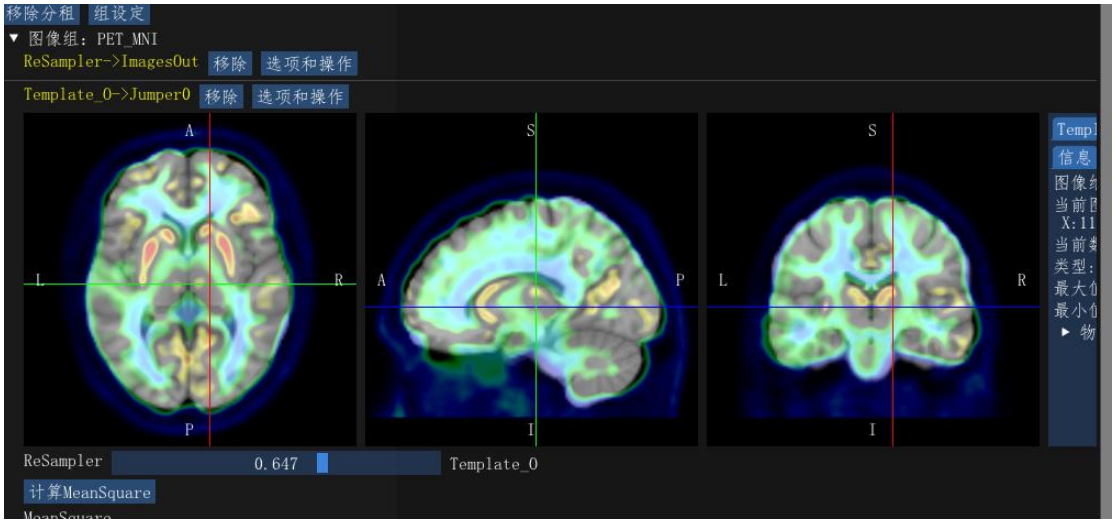


将 PET->T1 Affine 变换和 T1->MNI Affine 变换集合成一个复合变换。



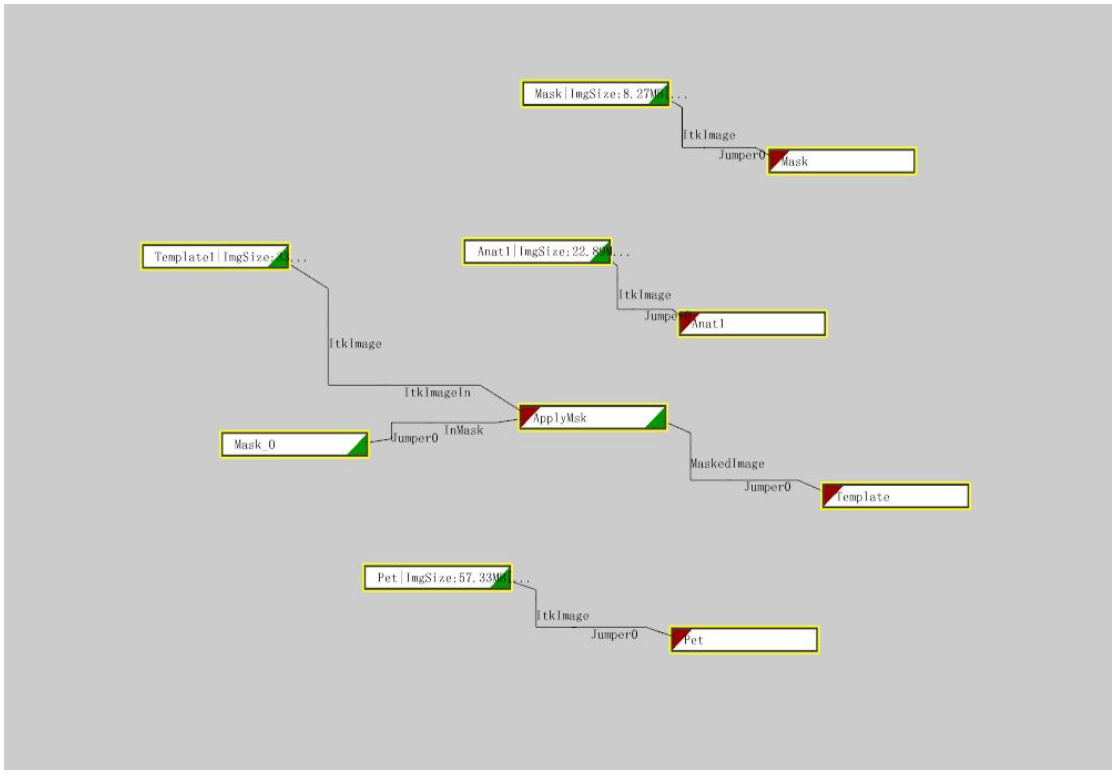
创建一个重采样器开启：输入变换并应用，输入变换场。

利用此采样器将 PET 图像重新采样到模板空间的同时，应用 PET->T1 Affine 变换、 T1->MNI Affine 变换、T1->MNI 的形变场，从而将 PET 图像一步 wrap 到 MNI 空间，在最小图像损失的前提下完成 PET 标准化。

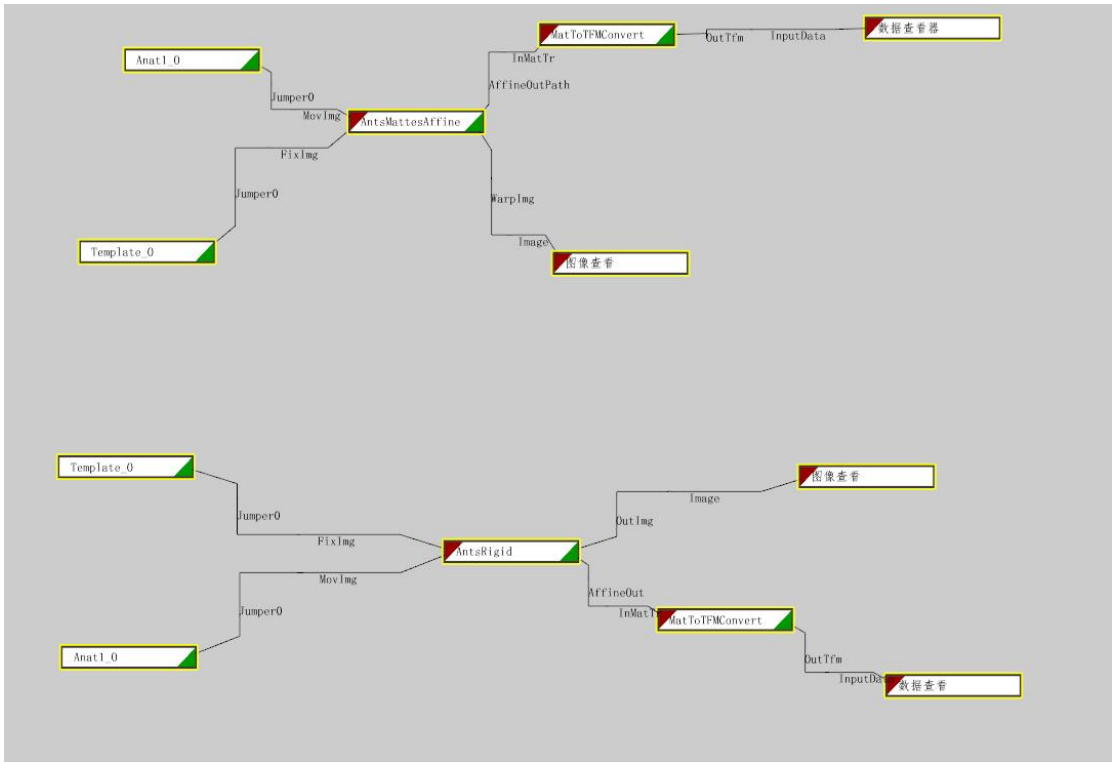


PET Wrap 结果

由于本图已经相对复杂且包含了多个数据流程，输入数据包括 T1，Template，PET，Mask。各个子图的设计过程中多次引用了输入数据，因此对设计图的输入部分创建输入跳跃器，再引用部分使用输出跳跃器，从而避免多次加载同一个图像。



加载后的图像直接传入输入跳跃器

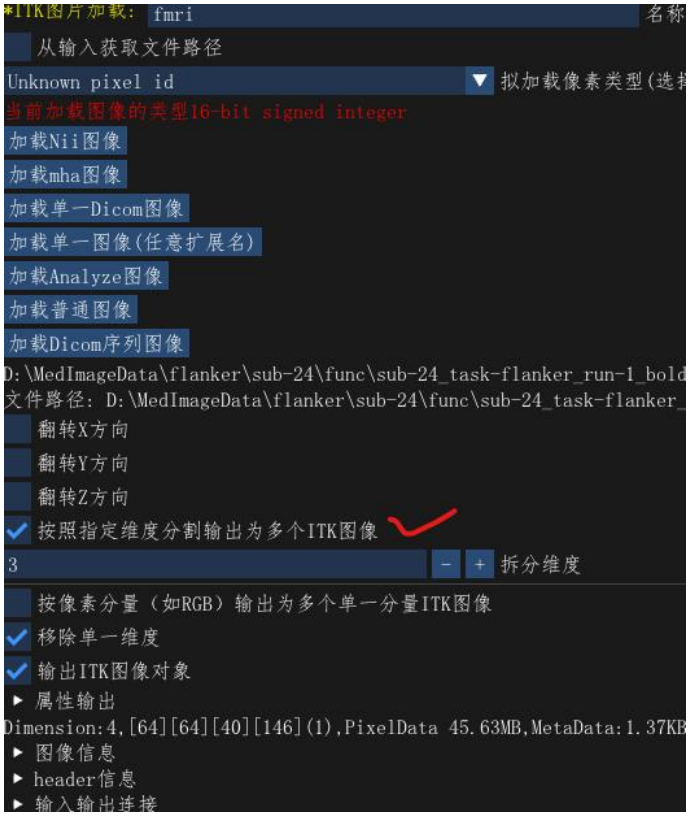


各个子图利用输出跳跃器引用输入数据

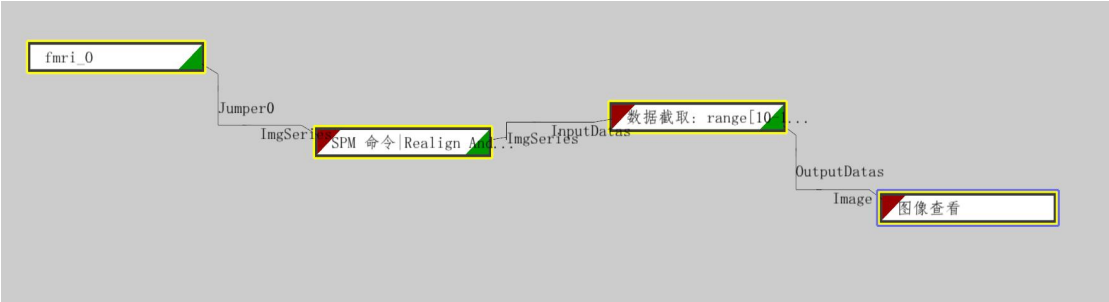
T1-FMRI Co-Register 到 MNI 模板

FMRI 时间序列影像协同 T1 配准到 MNI 模板

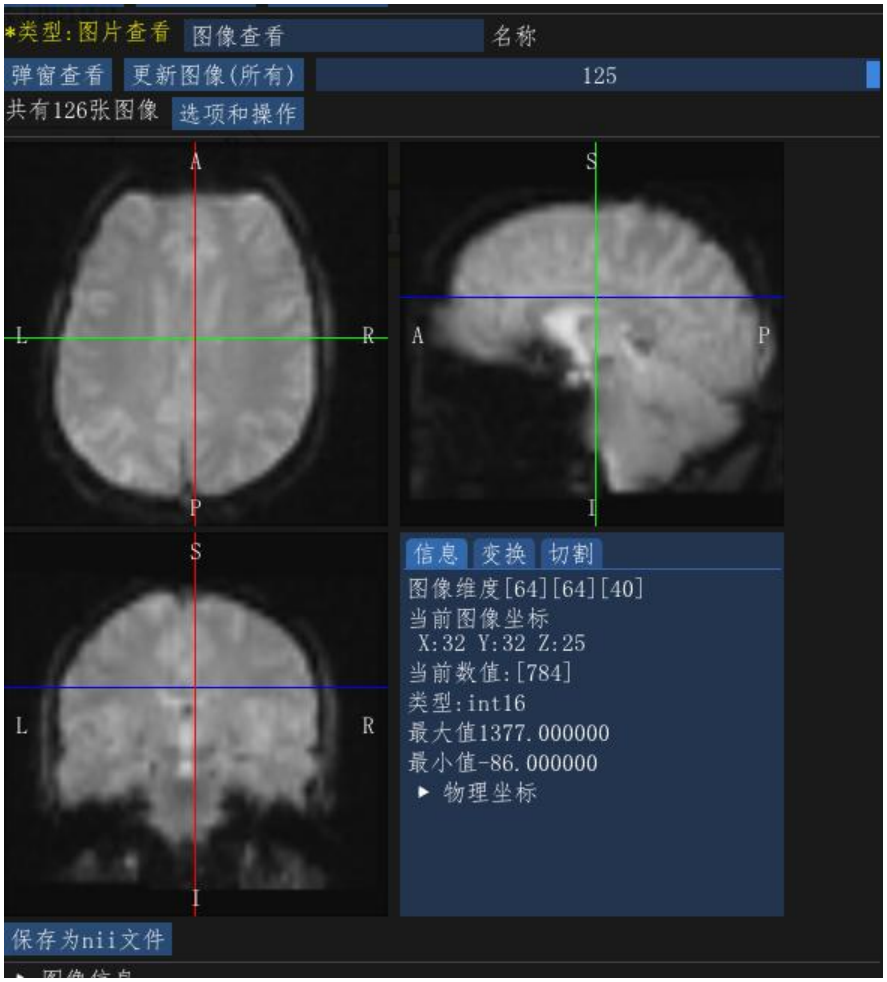
步骤 1： 加载 FMRI 并进行预处理



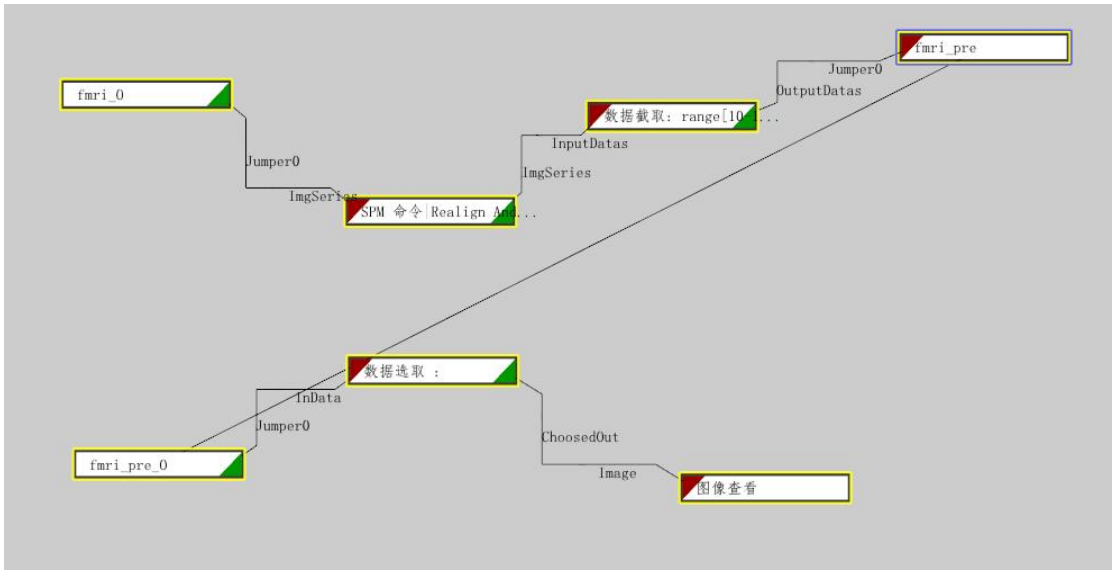
首先图像加载器选择“按照指定维度分割输出为多个影像”



SPM 命令节点 RealignAndReSlice 进行图像对齐后，数据截取节点取 10-136 帧数据，用于后续处理



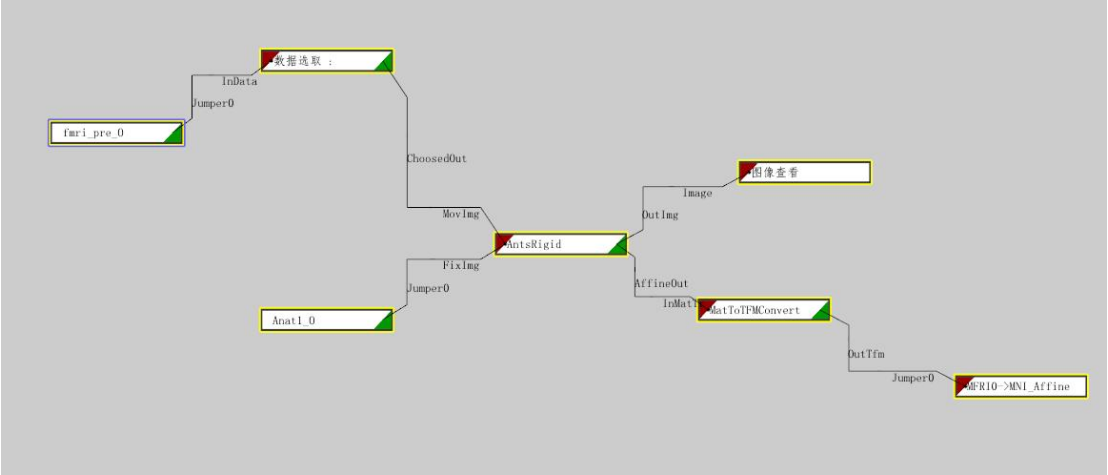
命令执行后得到 126 帧图像，头动矫正已经完成



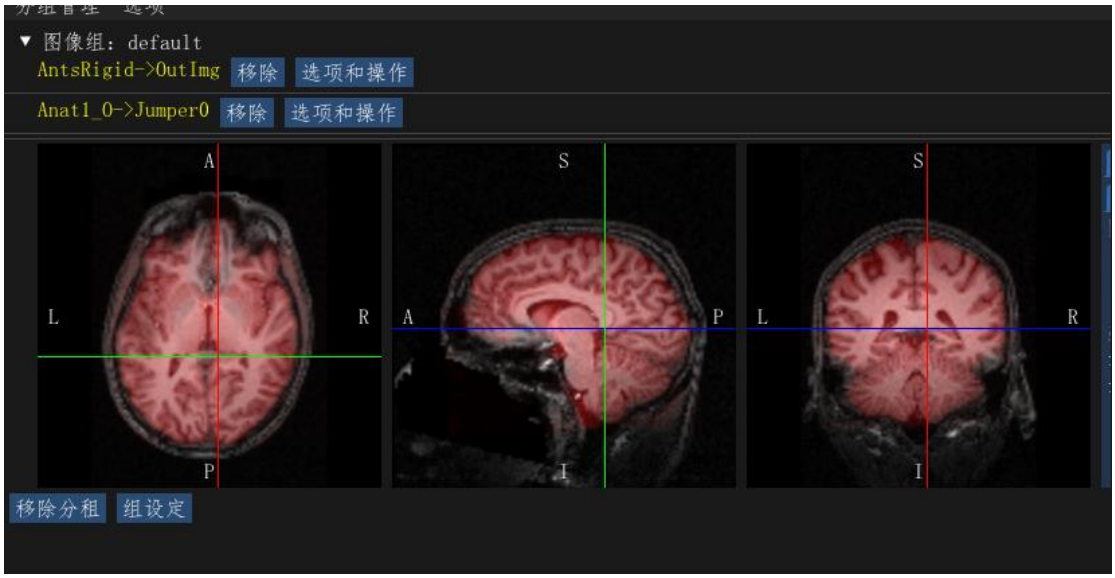
从处理好的图像列表中取出第一张（本例中图像加载时已经将 4D 图像分割为 3D 图像列表，否则需要利用 Extract Image Filter 取帧）

由于 FMRI 所有帧都已经对齐，所以在 0 帧（或者叫例帧）上所应用的所有变换都可以同步应用到同序列的其他帧上。0 帧以下简称 FMRI0。

步骤 2：FMRI0 刚体配准到 T1



配准变换同 PET->T1 配准，记录下仿射矩阵 FMRI0->MNI_AFFINE

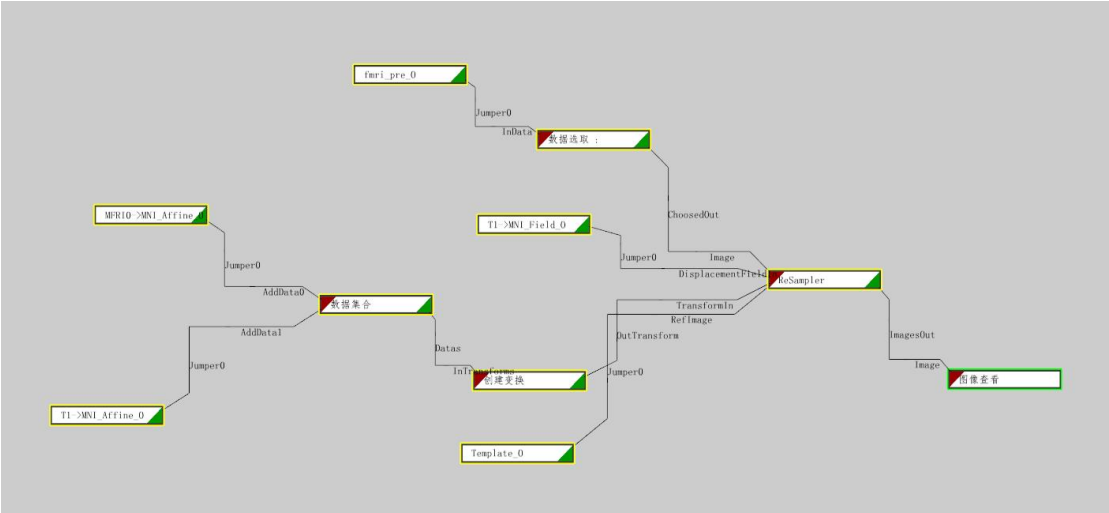


刚体配准结果良好

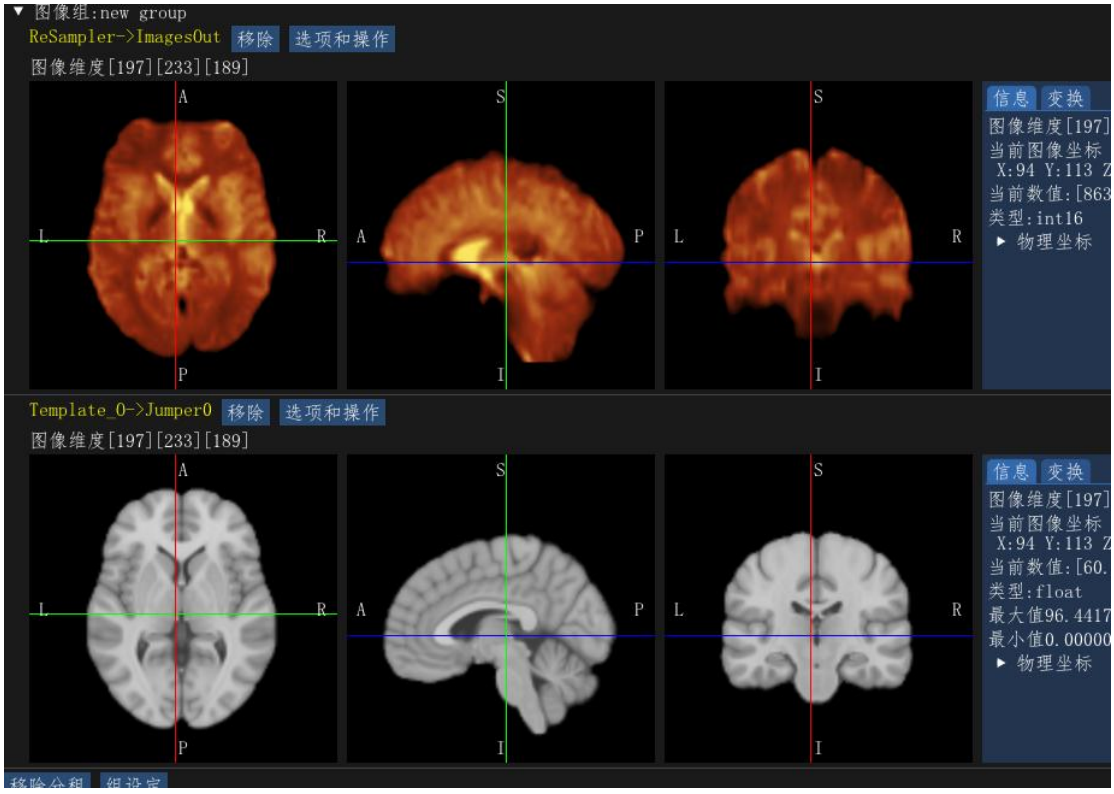
步骤 3：T1 配准到 MNI

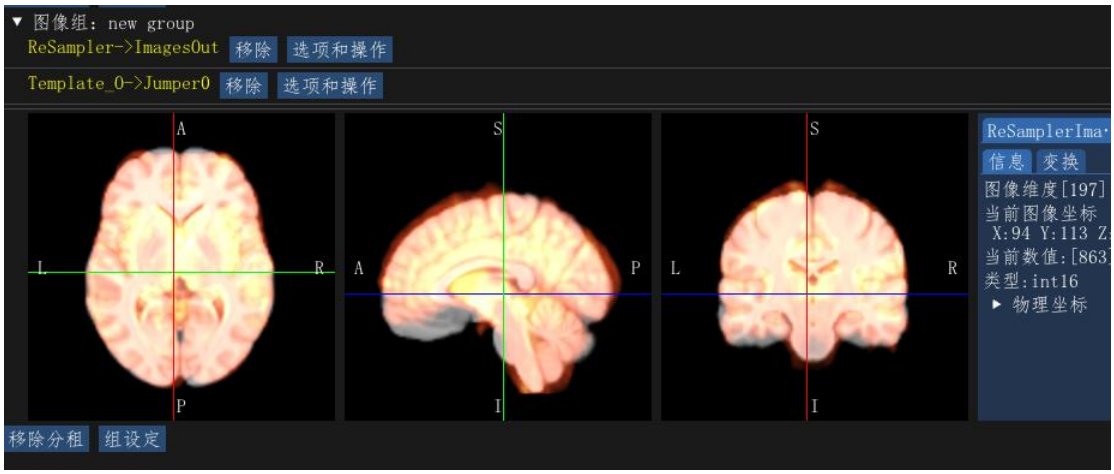
同上半部分不再赘述，保留 T1->MNI_Affine 和 T1->MNI_Field

步骤 4：将 FMRI0->T1、 T1->MNI_Affine、T1->MNI_Field 批量应用到时间序列上



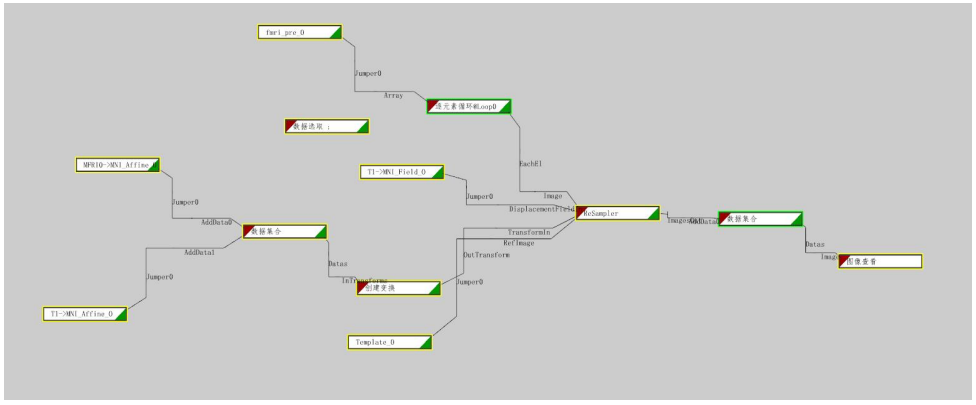
PET->MNI 案例类似一次采样将三个不同的变换依次应用到 FMRI0 图像上





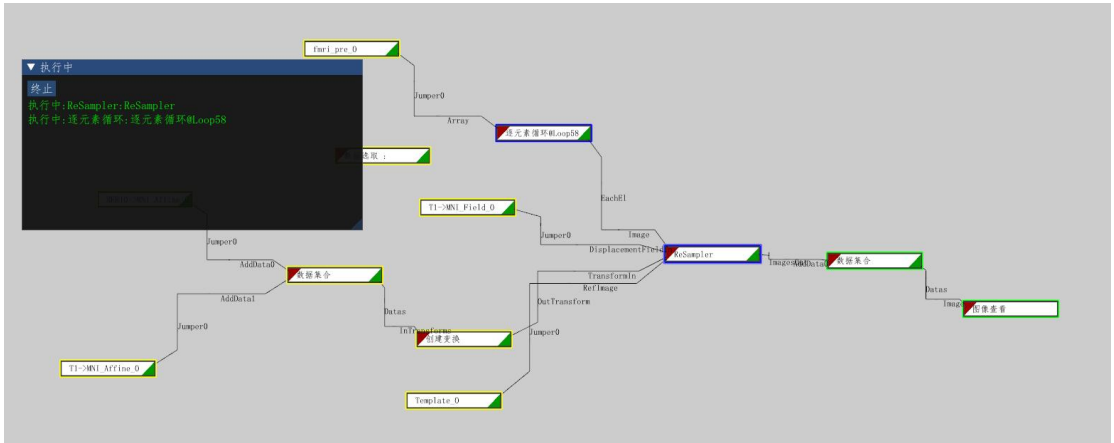
可以看到 FMRI0 已经和 MNI 模板配准

接下来批量应用到所有 126 帧 FMRI 上

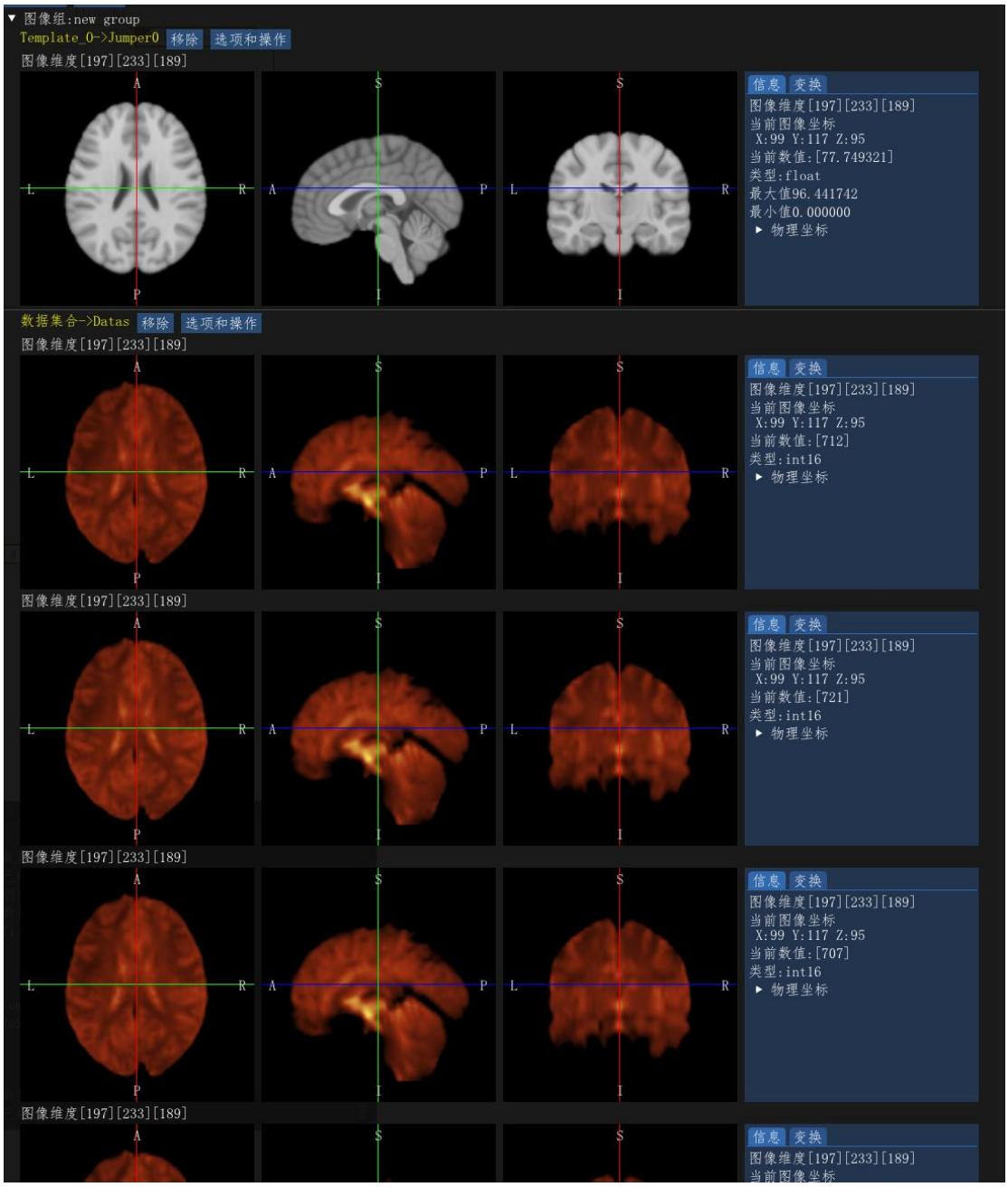


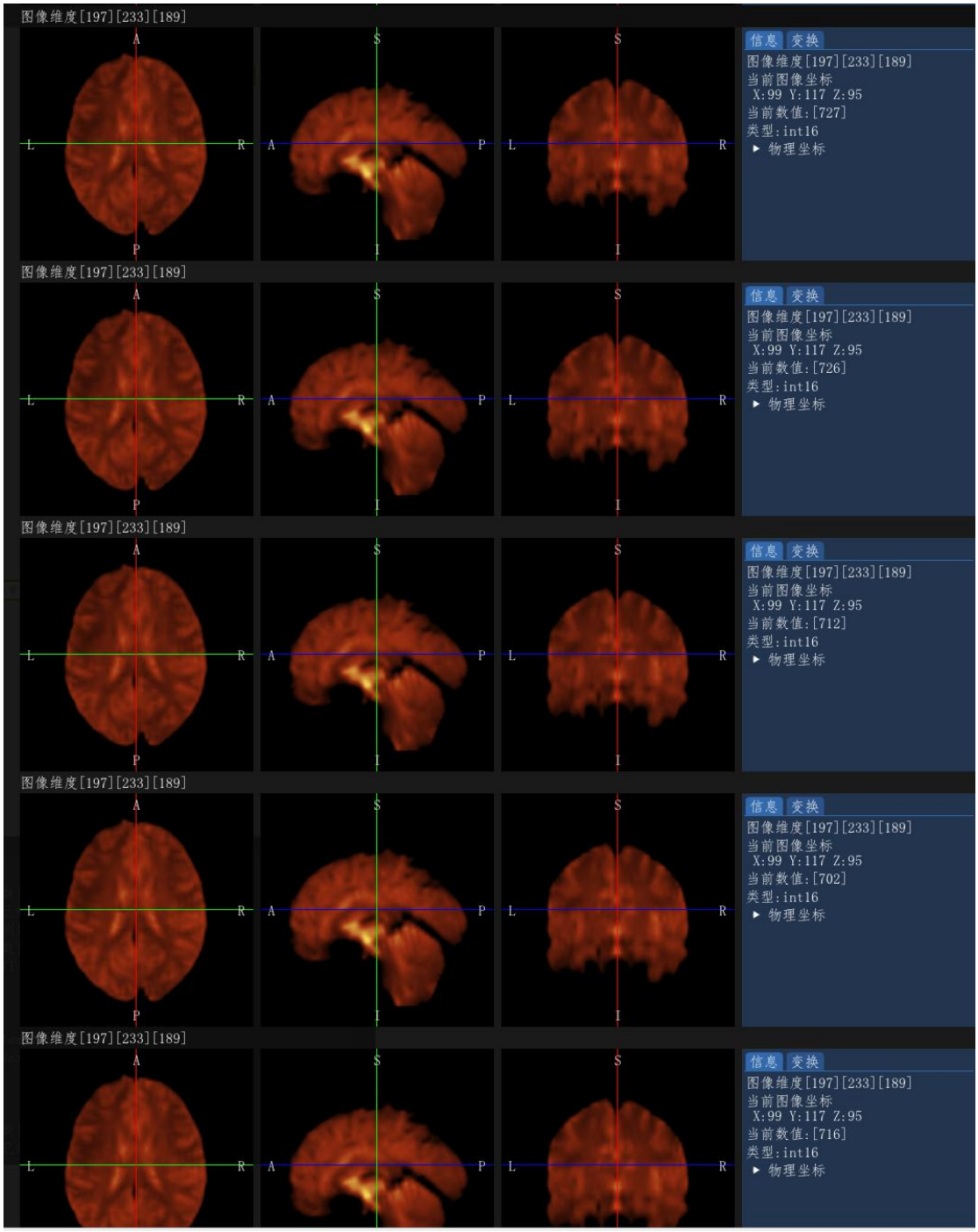
稍微修改上图增加一个元素循环节点，和数据集合节点，数据集合节点开启持续收集输入





循环处理每一帧数据


图像组:new group
Template_0->Jumper0
移除
选项和操作
图像维度[197][233][189]
信息
变换
图像维度[197][233][189]
当前图像坐标
X:99 Y:117 Z:95
当前数值:[77.749321]
类型:float
最大值96.441742
最小值0.000000
物理坐标
数据集合->Datas
移除
选项和操作
图像维度[197][233][189]
信息
变换
图像维度[197][233][189]
当前图像坐标
X:99 Y:117 Z:95
当前数值:[712]
类型:int16
物理坐标
图像维度[197][233][189]
信息
变换
图像维度[197][233][189]
当前图像坐标
X:99 Y:117 Z:95
当前数值:[721]
类型:int16
物理坐标
图像维度[197][233][189]
信息
变换
图像维度[197][233][189]
当前图像坐标
X:99 Y:117 Z:95
当前数值:[707]
类型:int16
物理坐标
图像维度[197][233][189]
信息
变换
图像维度[197][233][189]
当前图像坐标



可以看到所有 126 帧被正确 wrap 到 MNI 空间



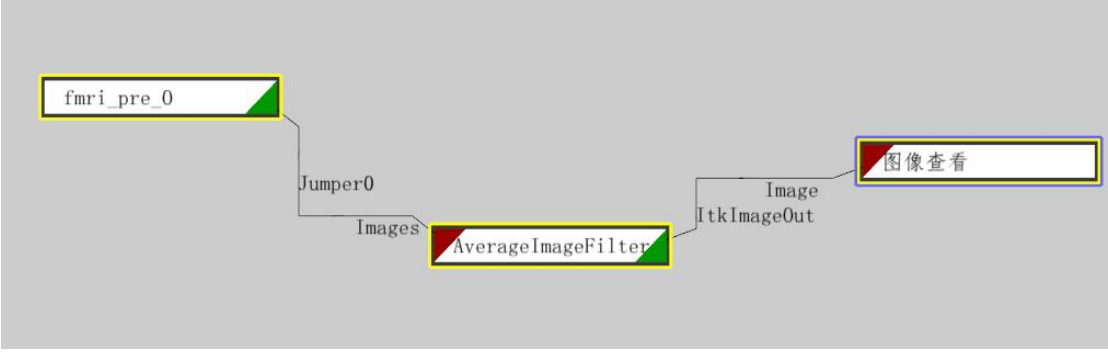
对图像数组 Overlay 每组只绘制当前图像选项开启后，可以通过滑动条 Imgs 选择某一帧对比查看

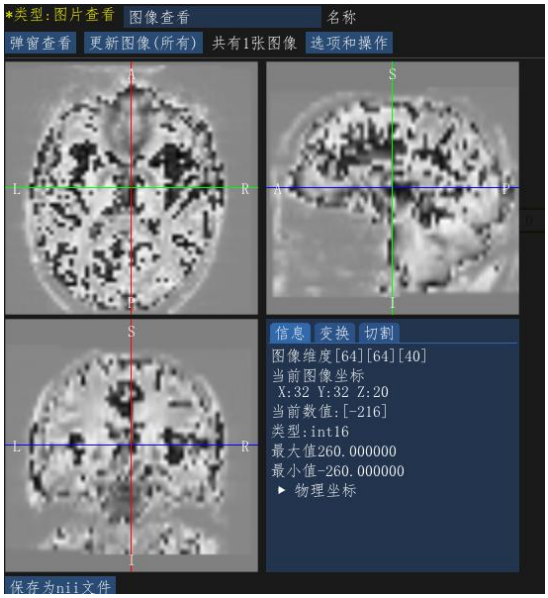
通过对比可以明确看出 FMRI 每一帧都正确 WRAP 到 MNI 空间下。

探讨和尝试

本例中 FMRI 的配准计算实际上只使用了一帧作为计算例（FMRI0），虽然配准计算之前已经对 FMRI 序列进行了对齐，但由于 FMRI 本身的图像特性较不稳定，FMRI0 帧的配准计算结果对于其他帧而言都存在可能的误差问题。因此显而易见的存在两种解决方案：

1. 每帧分别和 T1 刚体对齐再 wrap 到 MNI 空间。流程基本没有改变，只是增加了一个逐帧循环 Rigid 变换，这里不再赘述。
2. FMRI0 帧替换成 126 帧的平均图像，其余部分不变：

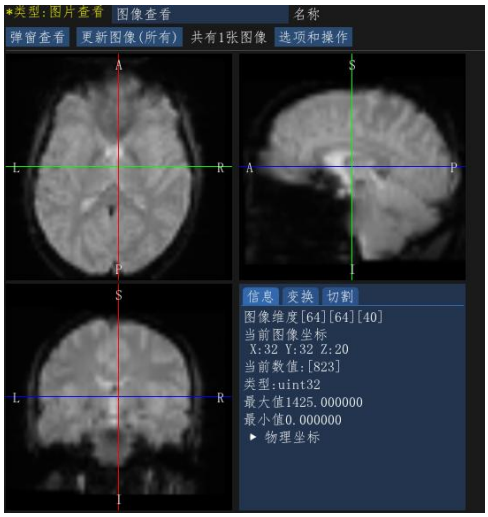




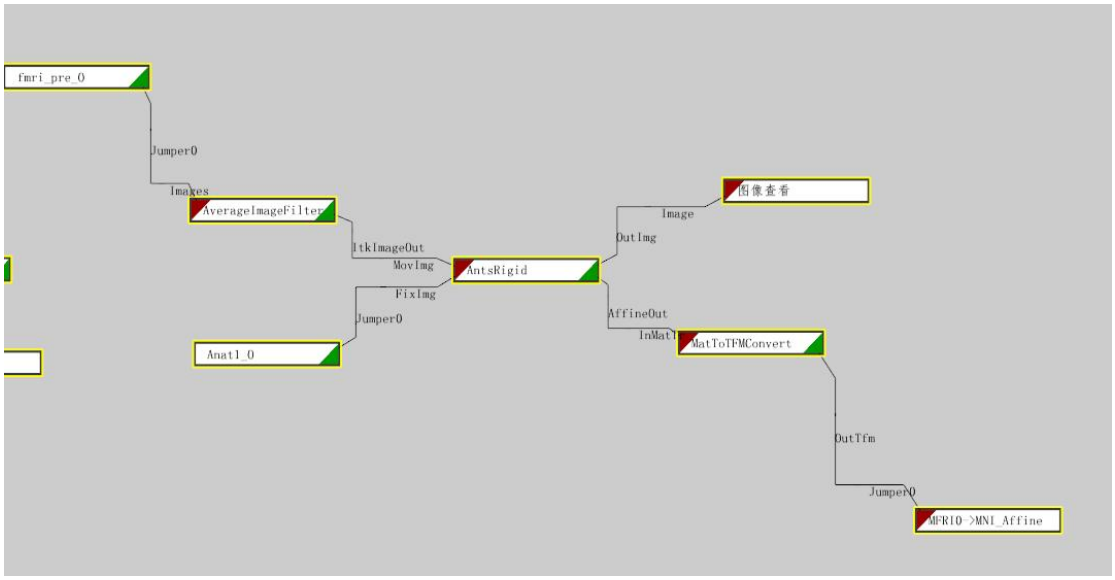
直接计算平均图像出现异常数值，原因是：图像数据类型无法保存计算中长的大数值。



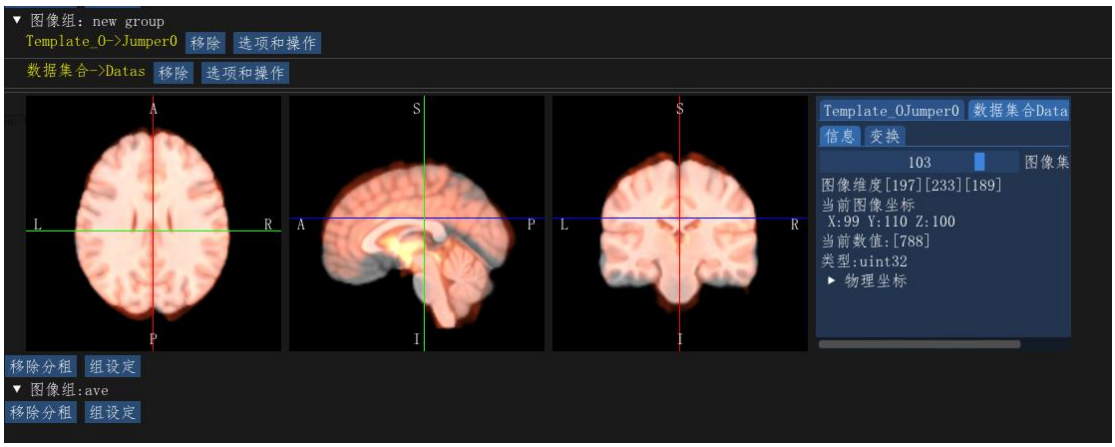
加载 fmri 时将数据类型改为 32-bit unsigned int



重新执行前面的步骤得到正确的均值图像



稍微修改 rigid 配准部分使用均值图像进行 To T1 Rigid 计算



最后执行的结果

如何对三种流程的结果进行对比从而确定哪种更加准确？如何设计计算流程来进行快速验证？