语音分离的总结

这篇语音分离（Speech Separation）的总结主要是为了分享一下目前语音分离领域的一些比较 work 的工作。该文章分为五个部分：

1. 纯语音分离任务的基本设定。
2. 纯语音分离任务的相关工作
3. 多模态的语音分离任务的基本设定
4. 多模态语音分离任务的相关工作
5. 纯语音分离和多模态语音分离任务的挑战

一、 纯语音分离任务的基本设定

该部分主要是为了介绍一下纯语音分离任务的基本流程和数据集的划分方式。首先说明一下什么是语音分离，如何来定义语音分离任务？语音分离的最终目的是将目标声音与背景噪声（环境噪声，人声等）进行分离。其实在这里语音分离还通常可以被人们称为“鸡尾酒会问题(cocktail party problem)”。此外，根据说话人（麦克风）的数目，我们通常将语音分离任务分为单通道（Single-channel)语音分离和麦克风阵列(Multi-channel)的语音分离。在这篇总结中，由于我个人前期的工作都是在做单通道语音分离，因此下文中的语音分离均为单通道语音分离任务。

对于语音分离任务我们通常的处理流程如下图所示。我们首先需要一个混合的语音信号，这个混合语音信号通常包含两到三个人的语音信号。然后，对于时频域的语音分离我们需要将时域的语音信号进行短时傅里叶变换（STFT），将时域信号转换为时频域信号。对于为什么需要进行 STFT，我个人的看法是对于时频域的信号特征更容易提取，更容易去做一些语音特征提取的操作，例如 MFCC 等。此外，对于经过 STFT 的时频域信号很容易的通过逆傅里叶变换（iSTFT）恢复为时域信号。同时，个人认为频域本质是把信号分解到每个子带空间上，每个空间里面性质稳定，你可以理解为频率恒定。因此，这解释了为什么一开始大家在做语音分离任务是都是在时频域上进行的。而对于时域的语音分离我们只需要做的就是搭建一个 encoder-decoder 的端到端的模型即可。

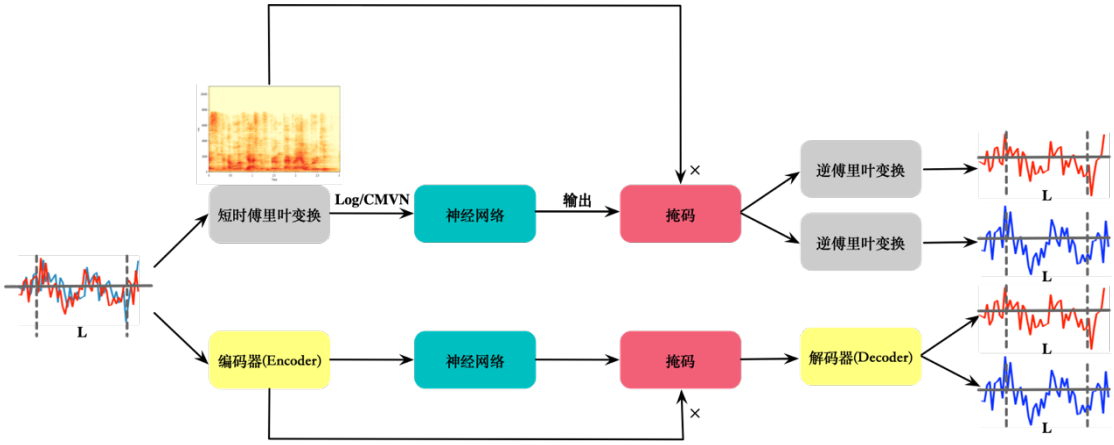


图 1. 语音分离的流程图

为什么语音分离任务在前期的进展比较缓慢，以及语音分离任务的难点是什么？首先我们可以通过流程图可以发现输入网络的往往只有一个标签，而输出却变成了两个标签。因为该任务数据监督学习，所以我们需要对应真实的标签来是

我们的模型进行学习。同时我们对训练的模型要求是说话者无关的模型(speaker independent)，就是我们训练了一个模型可以应用到所有说话人的语音分离。这就带来一个叫做标签置换问题（Label Permutation Problem)。就类似一个有固定个数个输出的系统(例如一个两个输出的神经网络)和三个待分离的说话人 A, B, C。在训练网络和计算损失函数的时候，两个输出需要人工指定一个顺序，这是网络输出的 permutation，一共有两种。由于我们不知道输出的 Permutation，因此就存在这个问题。现有的解决这个问题的方法基本分为两类，一类是 deep clustering 与其后续的 deep attractor network 等基于高维 embedding 的网络结构，另一类是 permutation invariant training 作为训练一般网络的方法。

对于语音分离任务，数据集一般使用的是华尔街日报数据集（Wall Street Journal dataset， WSJ0)。目前对于数据集的混合以及划分数据集的方法均采用 Deep Clustering 的 matlab 脚本混合（https://www.merl.com/demos/deep-clustering/create-speaker-mixtures.zip）。该数据集分为三个文件夹，si_tr_s，si_dt_05 和 si_et_05。训练数据集使用 si_tr_s 中获取，验证集和测试集均在 si_dt_05 和 si_et_05 中获取。在 Deep Clustering 方法之后所有的语音分离任务都是在开放演讲者集合（open speaker set）中训练，因此不会训练验证集和测试集的数据。

二、 纯语音分离任务的相关工作

在这个章节中我主要按照时间顺序来分享纯语音分离任务的经典工作。

2.1. 深度聚类（Deep Clustering）

在本方法中训练了一个深层网络[1]，将对比嵌入向量分配给频谱图的每个时频区域，以便根据输入混合物隐式预测目标频谱图的分段标签。这产生了一种基于深度网络的类似于频谱聚类的方法，因为这些嵌入形成了一个低秩的成对亲和力矩阵，该矩阵近似于理想的亲和力矩阵（亲和传播（AP）是基于数据点之间“消息传递”概念的聚类算法），同时能够实现更快的性能。该方法的目标函数是最大程度地减少由同一源控制的时频区域的嵌入之间的距离，同时最大化那些占主导地位不同的来源的时频区域的嵌入之间的距离。在测试时，聚类步骤通过针对未知分配优化 k-means 来“解码”嵌入中隐含的分割。

在音频源分离的情况下，这些区域被定义为每个源占主导的时频 bin 集合，估计这样的分区将使我们能够构建要应用于特征向量的时频掩码，从而导致时频表示，可以将其反转以获得孤立的信号源。其中，TF-bin：原始数据经过 FFT 后得到的频谱图表，频率轴的频率间阁或分辨率，通常取决采样率和采样点，如公式所示。

$$TF-Bin=\frac{采样率}{采样点数}$$

在该工作中，通过神经网络将时频域信号进行训练，然后根据构造的损失函数来使预测和目标信号的同一个 speaker 的 embedding 的 TF-bin 更接近，最终可以通过简单的聚类算法将不同说话者进行分开。在这里其实 Deep Clustering 规避了标签的匹配问题，他讲标签的匹配转换为亲和力矩阵的匹配程度。其实针对于亲和力矩阵的构造也存在着一定的问题，就是一个语音序列的 TF-Bin 非常的大，因此我们构造的亲和力矩阵也很大（TF×TF)，这样显存就不够用了。因此我们需要对构造方法做一个近似，将 TF-Bin 转换为 embedding 的维度 D，这

样就可以大大降低计算量。具体的变换和下面公式一致。

$$C_Y(V) = \|VV^T - YY^T\|_F^2 = \sum_{i,j}\left(\langle v_i, v_j\rangle - \langle y_i, y_j\rangle\right)^2$$

$$= \sum_{i,j:y_i=y_j}\left(|v_i - v_j|^2 - 1\right) + \sum_{i,j}\left(\langle v_i, v_j\rangle\right)^2$$

实现的 code 链接：https://github.com/JusperLee/Deep-Clustering-for-Speech-Separation

2.2. 帧级别和句子级别的标签不变训练（PIT&uPIT）

该方法是一种新颖的深度学习训练准则，称为置换不变训练（PIT）[2]。与深度聚类（DPCL）技术不同，该方法将分离误差直接最小化。其实就是直接找到最小标签排列组合的方法。

该方法首先使用深度学习模型估计一组掩码（mask），使用 softmax 操作可以轻松满足此约束。 然后，估计单一声音的频谱图，是通过 mask 点乘混合频谱图。该方法是第一个估计掩码的方法，因此可以优化模型参数以最小化估计掩码与理想比率掩码（IRM）之间的均方误差（MSE）。

$$J_m = \frac{1}{T \times F \times S}\sum_{s=1}^{S}\|\tilde{M} - M\|^2$$

这种方法有两个问题。首先，在静默段中，实际声音的 mask=0 且混合频谱图的 mask=0。所以他们就将 mask 之间的差异直接转为频谱图之间的差异。

$$J_m = \frac{1}{T \times F \times S}\sum_{s=1}^{S}\||\tilde{X}| - |X|\|^2$$

因为 PIT 是每个 meta-frame 独立做 PIT 决定，uPIT[3]是整个句子一起 sum over all frames 做决定。因此对于 uPIT 他的目标函数将变为一个 utterance 的 mse。

uPIT 的 code link:https://github.com/JusperLee/UtterancePIT-Speech-Separation

2.3. 深度吸引子网络（Deep Attractor Network）

DANet[4]首先是解决深度聚类的问题。为什么 DPCL 方法不好？首先他的目标函数与实际不符合。此外，DC 方法生成 mask 的时候是在 inference 部分，这部分是无法学习的，计算量很大，而且效果较差。而深度吸引子网络则是通过学习聚类中心来对不同的 speaker 生成不同的 mask。这样就可以得到一种可学习的聚类中心，与 DPCL 相比更加灵活，得到的结果也更加的理想。

DANet 的 code link：https://github.com/JusperLee/DANet-For-Speech-Separation

2.4 TasNet

TasNet[5]在时域上对音频进行分离，克服了短时傅里叶变换将时域信号转换为频域信号的精度问题。TasNet 使用编码器-解码器框架在时域中直接对信号

建模，并对非负编码器输出执行语音分离。这种方法省去了频率分解步骤，并将分离问题减少到估计编码器输出上的语音掩模，然后由解码器对其进行合成。TasNet 优于当前最新的因果和非因果语音（因果分离就是不考虑信号的将来特征，例如 RNN，LSTM，GRU。而非因果分离则是考虑信号的将来特征，例如，BILSTM，BIGRU，TCN）分离算法，降低了语音分离的计算成本，并显着降低了输出所需的最小延迟。TasNet 属于 encoder-decoder 框架，这种方法省去了 time-domain 转 frequency-domain 步骤，并将分离问题减少到 decoder 合成音频。该方法其实就是用卷积来替换 stft 方法，因为 stft 其实也是卷积操作。因果操作的性能不如非因果操作的的性能，这个是因为非因果操作可以考虑到将来的特征信息。

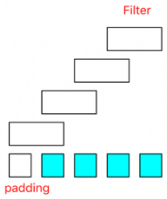


图 4. 因果卷积的 Padding

在训练阶段 TasNet 的目标函数是使尺度不变的信噪比（SI-SNR）最大化，该信噪比通常被用作替代标准信噪比的信源分离评估指标。从**这篇文章开始所有的时域方法**的损失函数均为该评价指标。SI-SNR 定义为：

$$S_{target} = \frac{\langle \hat{s}, s \rangle s}{\|s\|^2}$$

$$e_{noise} = \hat{s} - S_{target} \tag{9}$$

$$SI - SNR := 20\, log_{10} \frac{\|S_{target}\|}{\|e_{noise}\|}$$

2.5. Conv-TasNet

Luo Y 等人提出了一种全卷积的时域音频分离网络（Conv-TasNet[6]），这是一种用于端到端时域语音分离的深度学习框架。 Conv-TasNet 使用线性编码器来生成语音波形的表示形式，该波形针对分离单个 speaker 进行了优化。speaker 分离是通过将一组加权函数（掩码）应用于编码器输出来实现的。然后，使用线性解码器将修改后的编码器表示形式反转回波形。使用由堆叠的一维膨胀卷积块组成的时间卷积网络（TCN）查找掩码，这使网络可以对语音信号的长期依赖性进行建模，同时保持较小的模型尺寸。提出的 Conv-TasNet 系统在分离两个和三个 speakers 的混合物方面明显优于以前的时频掩蔽方法。

该方法与 TasNet 的区别就是将分离网络有循环神经网络（RNN，LSTM，GRU）变为时间卷积网络 TCN。这样可用通过 GPU 的并行加速大大缩减训练和测试时间。此外通过实验证明， Conv-TasNet 在因果实现和非因果实现中均大大提高了以前的 LSTM-TasNet 的分离精度。

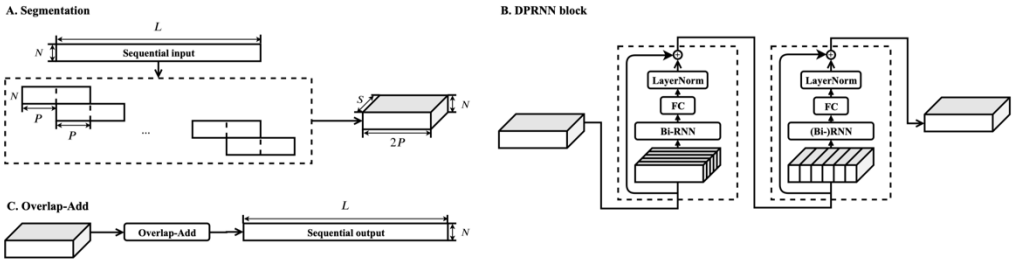
Tasnet 和 Conv-TasNet 的 code link：https://github.com/JusperLee/Conv-TasNet

2.6 Dual-Path-RNN

该文章的 Motivation 是如果出现超长的语音序列，使用传统的 RNN 模型将无法高效的处理(其原因是无法优化)。而一维卷积(1-D Conv)的感受野小于音频序列长度，因此无法进行 utterance-level 的语音分离。提出了一种双路径递归神经网络(DPRNN)[7]。该方法在深层模型中优化 RNN 使其可以对极长的语音序列建模。DPRNN 将较长的音频输入分成较小的块，并迭代应用块内和块间操作，其中输入长度（chunk size）可以与每个操作中原始音频长度的平方根成比例。

该方法的双路径的 pipeline 是首先将语音切分，然后组合成 3D 的特征快，学习块内特征和块间特征来分离语音。块内 RNN 首先独立处理本地块（将长语音序列分为 chunk number），然后块间 RNN 聚合来自所有块的信息做 utterance-level 的处理。

Dual-Path-RNN 的 code link：https://github.com/JusperLee/Dual-Path-RNN-Pytorch



2.7. 对于纯语音分离的实验结果

我们将不同模型的分离精度与使用 SDRi 和 SI-SNRi 的先前方法进行了比较。表 1 比较了纯语音模型在 WSJ0-2mix 数据集上其他最新方法的性能。对于所有系统，我们列出了文献中已报告的最佳结果。该表中缺少的值是因为没有在研究中报告该数字，或者是因为结果是使用不同的 STFT 配置计算的。先前 TasNet 用（B）LSTM-TasNet 表示。尽管 BLSTM-TasNet 的性能已超过 IRM 和 IBM，但无因果的 Conv-TasNet 在 SI-SNRi 和 SDRi 指标中的性能均大大超过了所有三个理想 T-F 掩模的性能，与所有以前的方法相比，模型尺寸都明显较小。而 Dual-Path-RNN 通过块内特征和块间特征的提取，最终获得了更好的结果。

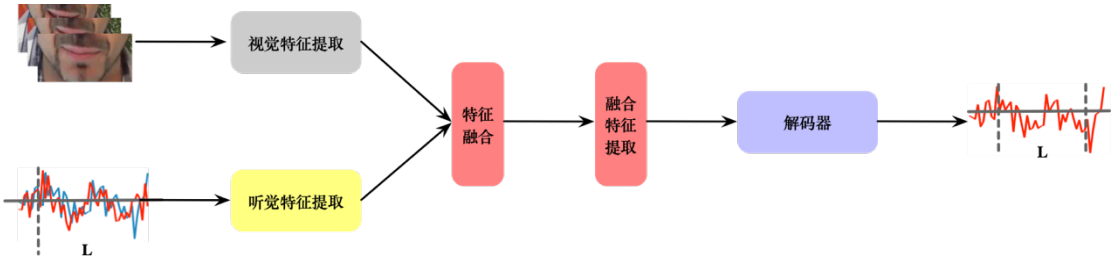
Method	Model size	Causal	SI-SNRi(dB)	SDBi(dB)
DPCL++	13.6M	×	10.8	-
uPIT-BLSTM-ST	92.7M	×	-	10.0
DANet	9.1M	×	10.5	-
ADANet	9.1M	×	10.4	10.8
cuPIT-Grid-RD	47.2M	×	-	10.2
CBLDNN-GAT	39.5M	×	-	11.0
Chimera++	32.9M	×	11.5	12.0
WA-MISI-5	32.9M	×	12.6	13.1
BLSTM-TasNet	23.6M	×	13.2	13.6

Conv-TasNet-gLN	5.1M	×	15.3	15.6
uPIT-LSTM	46.3M	√	-	7.0
LSTM-TasNet	32.0M	√	10.8	11.2
Conv-TasNet-cLN	5.1M	√	10.6	11.0
Dual-Path-RNN	2.6M	√	18.8	19.0
IRM	-	-	12.2	12.6
IBM	-	-	13.0	13.5
WFM	-	-	13.4	13.8

表 1. 与 WSJ0-2 MIX 数据集上的其他方法的比较。

三、视听模型的语音分离任务设定

视听模型的语音分离任务，由于其视觉信息的输入，将不会出现纯语音分离任务的的难题(标签置换问题)。这是由于我们输入的视觉信息是存在这标签的，与输出的标签数目是相同的。此外，视觉的特征信息用于将音频“聚焦”到场景中所需的说话者上，并改善语音分离质量。下图为视听模型的语音分离的 pipeline。

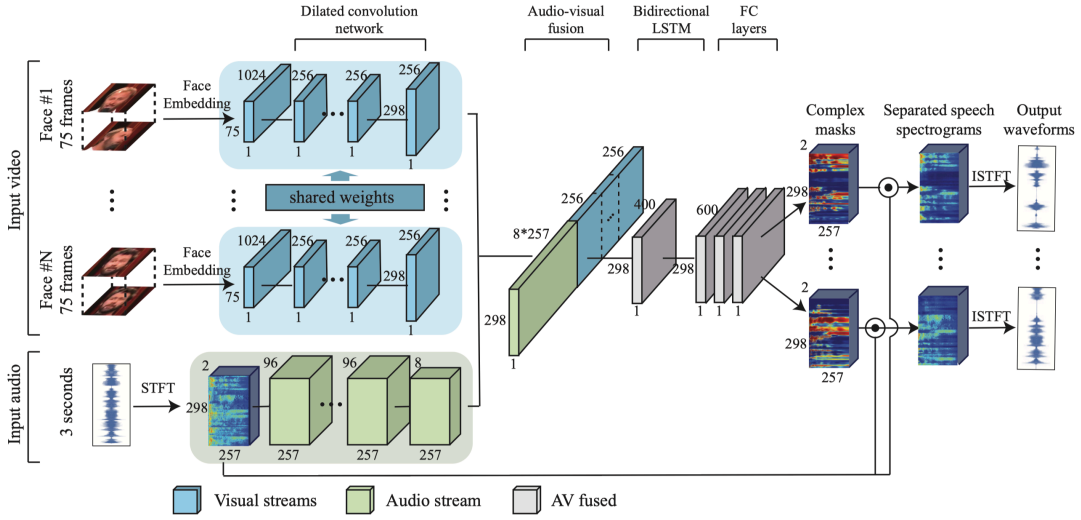


视听模型主要的数据集分别为 AVSpeech（https://looking-to-listen.github.io/avspeech/download.html），LRS2（http://www.robots.ox.ac.uk/~vgg/data/lip_reading/lrs2.html），VoxCeleb（www.robots.ox.ac.uk）。

四、视听模型的语音分离的相关工作

4.1. Looking to Listen at the Cocktail Party

Audio-only 的方法不能提供视频中说话者语音与视频之间的关联。AV-model 中的视觉功能用于将音频“聚焦”到场景中所需的说话者上，并改善语音分离质量。在混合语音的情况下，AV-model[8]显示出优于现有的纯音频语音分离的明显优势。另外，该方法是独立于说话者的（训练过一次，适用于任何说话者），比最近依赖于说话者的视听语音分离方法（要求针对每个感兴趣的说话者训练一个单独的模型）产生的效果更好。



输入：视频部分给定一个包含多个演讲者的视频片段，我们使用现成的面部检测器 mtcnn 在每个帧中查找面部 75 张。再使用 FaceNet 将人脸图像提取为一个 Face embedding。对面部图像的原始像素进行了实验，但这并没有提高性能。（每个扬声器共 75 张面部缩略图，假设以 25 FPS 播放 3 秒的片段）

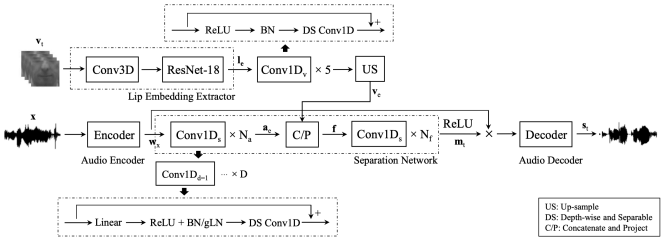
音频部分：每个时频（TF）箱包含复数的实部和虚部，将两者都用作输入。执行幂律压缩（power-law），以防止强的音频压倒弱的音频。对噪声信号和纯净参考信号都进行相同的处理。在实验中，分离模型可以应用于任意长的视频片段。当在一帧中检测到多个说话的脸部时，我们的模型可以接受多个脸部流作为输入。

输出：输出是一个乘法频谱 mask，它描述了 clean 语音与背景 noise 的时频关系。有研究人员观察到乘法掩模比其他方法（例如，频谱图幅度的直接预测或时域波形的直接预测）的工作效果更好。

该方法的 code link:https://github.com/JusperLee/Looking-to-Listen-at-the-Cocktail-Party

4.2. AV-Model

通过上述的研究可以发现一个结论，使用 CNN 的方法得到的结果往往要比使用 RNN，LSTM，GRU 的方法要更精确。这是由于 CNN 方法首先可以大大的缩小训练和测试时间，可以更好的利用 GPU。此外，由于 TCN 方法和 ResNet 方法的发展，CNN 网络也可以朝向更深层次发展，通过 Dilated Conv 操作可以使 CNN 也能获得序列的上下文语义，使 CNN 模型更能适合 RNN 网络的操作。因此 Conv-TasNet 可以得到更好的结果。所以我们在设计我们的 AV-model 的时候，语音部分参考了 Conv-Tasnet 的 Encoder 和 Separation 结构，而在视觉部分参考了端到端的 AudioVisual 语音识别的视觉信息部分。



视觉输入：视频编码器包含一个嘴唇嵌入提取器，然后是几个时间卷积块。嘴唇嵌入提取器包括一个 3D 卷积层和一个 18 层 ResNet，1-Dconv 每个块由一个时间卷积组成,之后是 ReLU 激活和批处理归一化。我们使用 256 维嘴唇嵌入，并为所有块选择 3 个内核大小，1 个步幅大小和 512 个通道大小。

音频输入：我们借鉴 Conv-Tasnet 的分离网络和编码器网络。同时我们设定 K 和 S 分别表示一维卷积运算中的内核大小和步幅。 在该模型中，我们默认使用 K = 40 和 S = 20。其中步幅大小相当于短时傅里叶变换中的窗移动大小。

融合部分：融合视觉特征的过程是通过在卷积通道尺寸上进行简单的串联操作来执行的，然后进行位置投影 P 以减小特征尺寸。 为了同步音频和视频功能的时间分辨率，如有必要，请在连接之前对视频流进行上采样。

五、 纯语音分离和多模态语音分离任务的挑战

目前，纯语音分离任务使用的数据集为 WSJ0。该数据集的特点是没有背景噪声，声音很清晰，且在录制的一段时间内没有停顿（即没有静音区的出现）。因此，通过该数据集得到的结果往往比较理想。但是我们在实际应用的情况往往不容乐观，我们录制的声音会夹夹杂环境噪声。同时，我们在录制的时候还有位置对于录制声音的强度也有一定的影响。此外，针对于含有静音区的音频，我们往往不能区分出该静音区的所属问题（例如 A 说了一段话暂停了一会儿，B 开始说，最后 A 和 B 一起说。在目前的所有语音分离方法都不能很好的去解决这个问题，这个意思不是不能分离，是不能将 A 所说的话全部分给 A，可能 A 说的话又分给了 B)。这个问题对于视听模型也是同样的问题。对于纯语音或者是多模态目前的研究还仅仅是针对于单通道的研究，麦克风阵列的语音分离目前正处于起步的阶段。首先是麦克风阵列的语音分离还没有被机器学习或者是深度学习给占领，大部分的方法还是一些滤波等算法。

而对于视听模型，我个人感觉还有很多要去研究或者是琢磨的地方。对于多模态的融合，我自己没有看过别的领域 multi-model 的方法，不太清楚怎么去做的融合。但是就语音领域大家的融合还是非常的单一，即将两个模态的输入 Concat 一下，然后送入一个融合的网络（CNN 或者 RNN 或者 Linear)。就感觉很草率粗暴的操作。所以感觉也需要深入的研究。然后，对于依赖视觉信息的输入的模型，如果不能准确地识别出嘴唇或者在录像时目标人突然捂住嘴唇，也会对分离的准确性产生一定的影响，这个可能也是未来的一个切入点。还有一个我觉得可能会有用的是关于位置信息的，就是如果是麦克风阵列的分离或者增强，我们如果在输入的视觉信息考虑其深度信息（位置信息)，是否也可以对结果产生正向的影响。这个以后通过实验来证实吧，毕竟 CV 领域大家对深度信息提取肯定也研究的比较多。最后，我想说的是以上是我自己在这半年多时间学习的的一些内容的总结，可能也有一定的问题或者错误，欢迎指正。

六、 参考文献

1. Hershey, J., Chen, Z., Roux, J., Watanabe, S. (2016). Deep clustering: Discriminative embeddings for segmentation and separation 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)
2. Yu, D., Kolbaek, M., Tan, Z., Jensen, J. (2017). Permutation invariant training of deep models for speaker-independent multi-talker speech separation 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)
3. Kolbæk, M., Yu, D., Tan, Z., Jensen, J. (2017). Multi-talker Speech Separation with Utterance-level Permutation Invariant Training of Deep Recurrent Neural Networks
4. Chen, Z., Luo, Y., Mesgarani, N. (2016). Deep attractor network for single-microphone speaker separation
5. Luo, Y., Mesgarani, N. (2018). TaSNet: Time-Domain Audio Separation Network for Real-Time, Single-Channel Speech Separation 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)
6. Luo, Y., Mesgarani, N. (2018). Conv-TasNet: Surpassing Ideal Time-Frequency Magnitude Masking for Speech Separation IEEE/ACM transactions on audio, speech, and language processing 27(8), 1256-1266.
7. Luo, Y., Chen, Z., Yoshioka, T. (2019). Dual-path RNN: efficient long sequence modeling for time-domain single-channel speech separation
8. Ephrat, A., Mosseri, I., Lang, O., Dekel, T., Wilson, K., Hassidim, A., Freeman, W., Rubinstein, M. (2018). Looking to Listen at the Cocktail Party: A Speaker-Independent Audio-Visual Model for Speech Separation ACM Transactions on Graphics (TOG) 37(4), 112.
9. Wu, J., Xu, Y., Zhang, S., Chen, L., Yu, M., Xie, L., Yu, D. (2019). Time Domain Audio Visual Speech Separation 2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)