Prof. Dr.-Ing. Timo Gerkmann

Machine Learning for Speech Signal Processing on Hearing Devices

Universität Hamburg
Department of Informatics
Signal Processing (SP)
August 13, 2022
How can Machine Learning help to make information more easily accessible by humans and machines.
1. Single Channel Source Separation

2. Phase Estimation Enables High Quality at Low Latency

3. Non-linear Multi-channel Filtering

Single Channel Source Separation
Cocktail-Party Problem

Conditions:
- Undefined number of speakers
- Unknown speakers
- Single microphone

- Video captioning
- Meeting transcription
- Hearing aids
- ...

Conclusions

- Machine Learning enables separating sources recorded with only one microphone
- As traditional approaches, these algorithms can be made real-time capable
- The algorithmic latency depends on the chosen frame sizes
Phase Estimation Enables High Quality at Low Latency

Tal Peer, M.Sc.
STFT-based speech processing traditionally uses frames of around 32ms

- Short enough to capture non-stationarity of speech
- Long enough to admit a reasonable spectral resolution

Is this optimal?
- Frame length imposes a lower bound on algorithmic latency
- The justification for 32ms is mainly based magnitude and ignores phase

As traditional enhancement methods are magnitude centric, 32ms appears a well motivated choice
- But what if we were able to also estimate phase?

- Kazama et al.\cite{3}: listening experiment on intelligibility under variation of frame length

→ The information contained in magnitude and phase varies with frame length

---

Phase, Magnitude and Frame Length (2)

![Graph showing the relationship between Speech Intelligibility in % and Frame Length in ms for different phase, magnitude, and frame length settings. The graph compares MSS and PSS with various frame lengths, demonstrating peaks and troughs in intelligibility.]
Medium frames: magnitude suffices for good reconstruction

Short and long frames: magnitude loses relevance, good reconstruction is possible from phase alone
Phase information gets important for short frames.

Model-based phase estimation methods exist only for long frames\[^{4,5}\].

**Research Questions**

- Can we use modern machine learning approaches to estimate phase when using short frames?
- Which frame length for phase-aware STFT-based networks?


Evaluation

We use a DNN with explicit magnitude and phase estimation\cite{6,7} to

- Quantify the contribution of phase and magnitude estimation for different frame lengths
- Quantify overall performance of joint network for different frame length


Results

✔ Trend observed on oracle data carries over to DNN-based magnitude and phase estimation
✔ Machine Learning can be used to estimate phase also with short frames
✔ Phase-aware processing is particularly beneficial with short frames
✔ Short frames of 4ms enable short latency and high quality
Non-linear Multi-channel Filtering

Kristina Tesch, M.Sc.
Multi-channel Speech Enhancement

- Analytic solutions
- Computationally lightweight
- MMSE optimal for Gaussian noise\(^8\)

- Drops linearity assumption
- Integrates spatial and tempo-spectral processing
  - More powerful processing model
  - Parameter estimation challenging

Proof of Concept with Oracle Data

Inhomogeneous noise field created by five directional Gaussian noise sources

Joint non-linear spatial-spectral filter is a more powerful processing model

Δ POLQA: 2.64 ± 0.08
Δ SI-SDR: 9.92 ± 0.30
We saw: Joint nonlinear spatial spectral filtering (JNF) is more powerful than traditional beamformer + postfilter

Above examples provided a proof of concept, but estimating the required higher-order statistics is very difficult in practice

---

We saw: Joint nonlinear spatial spectral filtering (JNF) is more powerful than traditional beamformer + postfilter.

Above examples provided a proof of concept, but estimating the required higher-order statistics is very difficult in practice.

Research Questions

- Do our theoretical findings carry over when learning a JNF using DNNs?
  - Such DNN-based JNFs are fundamentally different to *DNN-guided* beamformers!

- How important are the interdependencies between different sources of information?

- What are the implications that arise for the design of network architectures?

---

Network structure that allows to easily **control the integration of different sources of information**

- Combine spatial with spectral (F-JNF) or temporal (T-JNF) information or both (FT-JNF)
Approach - Dataset

Speaker extraction focusing on spatial filtering capabilities

Task:
- Speaker extraction scenario
- 2-5 microphones in a circular array
- 1 target speaker
- 5 interfering speakers

Dataset generation:
- Clean speech from WSJ0 (75 male and 74 female speakers)
- 6000 training examples (25 hours of training data)
- Simulation using the source-image model
- SNR between -9 and 2 dB
- Room dimensions between (2.5 × 3 × 2.2) and (5 × 9 × 3.5) meters
- T60: 0.2 – 0.5 seconds
Interdependency Between Information Sources

<table>
<thead>
<tr>
<th></th>
<th>Δ POLQA</th>
<th>ESTOI</th>
</tr>
</thead>
<tbody>
<tr>
<td>F-JNF</td>
<td>1.15</td>
<td>0.70</td>
</tr>
<tr>
<td>T-JNF[10]</td>
<td>0.74</td>
<td>0.63</td>
</tr>
<tr>
<td>FT-JNF (ours)</td>
<td>1.43</td>
<td>0.76</td>
</tr>
</tbody>
</table>

- Additional spectral information is more valuable than temporal information.
- Spectral information increases the spatial selectivity.

Non-Linear Versus Linear Spatial Filter

Blind estimation using DNNs

A joint non-linear filter outperforms an *oracle* linear spatial filter plus post-filter.
Comparison with State-of-the-art Methods

- Complex masked-based: FT-JNF, T-JNF, CRNN
- Beamformer-inspired: FaSNet+TAC, EaBNet, COSPA
- FT-JNF and T-JNF have the same lowest number of parameters

→ Proposed FT-JNF outperforms all other methods

---


Conclusions

Deep non-linear filters overcome the linear processing model and exploit dependencies between spatial and tempo-spectral information.

- Spectral information increases the spatial selectivity of the filter.
- The proposed scheme that exploits spatial, spectral and temporal information outperforms state-of-the-art network architectures.
Diffusion-based Generative Models for Speech Enhancement

Julius Richter, M.Sc. and Simon Welker, M.Sc.
Speech enhancement algorithms are nowadays dominated by the use of deep neural networks (DNNs)

- Exploit temporal-spectral structure to distinguish speech from noise

### Discriminative model vs. generative model

- Statistical models can be classified as generative or discriminative
- Discriminative models dominate the task of speech enhancement
- Recently, there is a trend towards generative approaches
Why Generative Modeling?

<table>
<thead>
<tr>
<th>Discriminative models</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Learn to directly map noisy speech to the corresponding clean speech</td>
</tr>
<tr>
<td>- Trained with a variety of clean/noisy speech pairs</td>
</tr>
<tr>
<td>✖ No guarantee of robustness in unseen situations</td>
</tr>
<tr>
<td>✖ Unpleasant speech distortions may outweigh the benefits of noise reduction</td>
</tr>
</tbody>
</table>
## Why Generative Modeling?

### Discriminative models
- Learn to directly map noisy speech to the corresponding clean speech
- Trained with a variety of clean/noisy speech pairs
  - No guarantee of robustness in unseen situations
  - Unpleasant speech distortions may outweigh the benefits of noise reduction

### Generative models
- Learn a prior distribution over clean speech data
- Infer clean speech from noisy signals that are assumed to lie outside the learned distribution
  - Generalize well to unseen acoustic situations
  - Aim to produce natural sounding speech
Popular deep generative models

- Variational autoencoders (VAEs)
- Generative adversarial networks (GANs)
- Auto-regressive models
- Diffusion-based generative models
Our contributions:

- Incorporate temporal dependencies into the VAE\textsuperscript{[15]}
- Improve the robustness with a noise-aware encoder\textsuperscript{[16]}
- Guide the VAE with a supervised classifier trained on voice activity or ideal binary mask prediction\textsuperscript{[17]}
- Disentanglement learning of the latent variables applied to audio-visual voice activity detection\textsuperscript{[18]}

Limitations:

- VAE needs additional noise estimator to form a Wiener filter
- Limited by the bottleneck of the latent representation


Generative diffusion models\cite{19,20} consist of two processes:

- Forward diffusion process that gradually adds noise to the input
- Reverse process that learns to generate data by denoising


Stochastic Diffusion Process

- Model the corruption of clean speech as a diffusion process \( \{x_t\}_{t=0}^T \) \[21\]
- Define the diffusion process as a solution to a stochastic differential equation (SDE) \[22\]

\[
dx_t = \gamma(y - x_t)dt + g(t)d\mathbf{w}
\]

The learned score model \( s_\theta(x_t, y, t) \) predicts the added Gaussian noise

---


Reverse Sampling

- Initialize reverse process with $\tilde{x}_T \sim \mathcal{N}_C(x_T; y, \sigma(T)^2 I)$
- Solve reverse SDE with general-purpose SDE solvers (e.g. Euler-Maruyama)
- Necessary reverse steps $N \approx 30 \Rightarrow 30$ model calls

$$d\tilde{x}_t = [-\gamma(y - \tilde{x}_t) + g(t)^2 s_\theta(x_t, y, t)] dt + g(t)d\tilde{w}$$
Datasets:

- **WSJ0-CHiME3**
  - Clean speech utterances from Wall Street Journal (WSJ0)[23]
  - Noise signals from CHiME3[24]
  - SNR uniformly sampled between 0 and 20 dB
- **Voicebank-Demand[25]**
  - Standardized dataset often used as a benchmark

Matched and mismatched conditions:

<table>
<thead>
<tr>
<th>condition</th>
<th>train/valid</th>
<th>test</th>
</tr>
</thead>
<tbody>
<tr>
<td>matched</td>
<td>WSJ0-CHiME3</td>
<td>WSJ0-CHiME3</td>
</tr>
<tr>
<td>mismatched</td>
<td>Voicebank-Demand</td>
<td>WSJ0-CHiME3</td>
</tr>
</tbody>
</table>


### Matched condition: Training and test on same datasets

<table>
<thead>
<tr>
<th>Method</th>
<th>Type</th>
<th>POLQA</th>
<th>ESTOI</th>
<th>SI-SDR [dB]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mixture</td>
<td>-</td>
<td>2.63 ± 0.67</td>
<td>0.78 ± 0.14</td>
<td>10.0 ± 5.7</td>
</tr>
<tr>
<td>RVAE</td>
<td>G</td>
<td>2.97 ± 0.63</td>
<td>0.85 ± 0.11</td>
<td>15.8 ± 5.0</td>
</tr>
<tr>
<td>CDiffuSE</td>
<td>G</td>
<td>2.77 ± 0.52</td>
<td>0.80 ± 0.09</td>
<td>7.3 ± 1.9</td>
</tr>
<tr>
<td>SGMSE+ (ours)</td>
<td>G</td>
<td>3.71 ± 0.50</td>
<td>0.92 ± 0.05</td>
<td>17.2 ± 4.6</td>
</tr>
<tr>
<td>Conv-TasNet</td>
<td>D</td>
<td>3.65 ± 0.54</td>
<td><strong>0.93 ± 0.05</strong></td>
<td><strong>19.9 ± 4.3</strong></td>
</tr>
<tr>
<td>MetricGAN+</td>
<td>D</td>
<td>3.52 ± 0.61</td>
<td>0.88 ± 0.08</td>
<td>10.5 ± 4.5</td>
</tr>
</tbody>
</table>
## Results

**Matched condition:** Training and test on same datasets

<table>
<thead>
<tr>
<th>Method</th>
<th>Type</th>
<th>POLQA</th>
<th>ESTOI</th>
<th>SI-SDR [dB]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mixture</td>
<td>-</td>
<td>2.63 ± 0.67</td>
<td>0.78 ± 0.14</td>
<td>10.0 ± 5.7</td>
</tr>
<tr>
<td>RVAE</td>
<td>G</td>
<td>2.97 ± 0.63</td>
<td>0.85 ± 0.11</td>
<td>15.8 ± 5.0</td>
</tr>
<tr>
<td>CDiffuSE</td>
<td>G</td>
<td>2.77 ± 0.52</td>
<td>0.80 ± 0.09</td>
<td>7.3 ± 1.9</td>
</tr>
<tr>
<td>SGMSE+ (ours)</td>
<td>G</td>
<td>3.71 ± 0.50</td>
<td>0.92 ± 0.05</td>
<td>17.2 ± 4.6</td>
</tr>
<tr>
<td>Conv-TasNet</td>
<td>D</td>
<td>3.65 ± 0.54</td>
<td>0.93 ± 0.05</td>
<td>19.9 ± 4.3</td>
</tr>
<tr>
<td>MetricGAN+</td>
<td>D</td>
<td>3.52 ± 0.61</td>
<td>0.88 ± 0.08</td>
<td>10.5 ± 4.5</td>
</tr>
</tbody>
</table>

**Mismatched condition:** Training and test on different datasets

<table>
<thead>
<tr>
<th>Method</th>
<th>Type</th>
<th>POLQA</th>
<th>ESTOI</th>
<th>SI-SDR [dB]</th>
</tr>
</thead>
<tbody>
<tr>
<td>RVAE</td>
<td>G</td>
<td>2.84 ± 0.61</td>
<td>0.82 ± 0.11</td>
<td>13.9 ± 4.8</td>
</tr>
<tr>
<td>CDiffuSE</td>
<td>G</td>
<td>2.20 ± 0.50</td>
<td>0.71 ± 0.10</td>
<td>3.8 ± 2.5</td>
</tr>
<tr>
<td>SGMSE+ (ours)</td>
<td>G</td>
<td>3.43 ± 0.61</td>
<td>0.90 ± 0.07</td>
<td>16.2 ± 4.1</td>
</tr>
<tr>
<td>Conv-TasNet</td>
<td>D</td>
<td>3.13 ± 0.60</td>
<td>0.88 ± 0.08</td>
<td>15.2 ± 3.9</td>
</tr>
<tr>
<td>MetricGAN+</td>
<td>D</td>
<td>2.47 ± 0.67</td>
<td>0.76 ± 0.12</td>
<td>6.8 ± 3.1</td>
</tr>
</tbody>
</table>
- Performance for the matched condition is on par or even slightly better than discriminative baselines

✔ Proposed approach is more robust in unseen situations
10 participants rate 12 random examples from the test set
### Audio Examples

<table>
<thead>
<tr>
<th></th>
<th>Matched</th>
<th>Mismatched</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clean</td>
<td><img src="audio" alt="Play" /></td>
<td><img src="audio" alt="Play" /></td>
</tr>
<tr>
<td>Noisy</td>
<td><img src="audio" alt="Play" /></td>
<td><img src="audio" alt="Play" /></td>
</tr>
<tr>
<td>RVAE</td>
<td><img src="audio" alt="Play" /></td>
<td><img src="audio" alt="Play" /></td>
</tr>
<tr>
<td>SGMSE+</td>
<td><img src="audio" alt="Play" /></td>
<td><img src="audio" alt="Play" /></td>
</tr>
<tr>
<td>Conv-TasNet</td>
<td><img src="audio" alt="Play" /></td>
<td><img src="audio" alt="Play" /></td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th></th>
<th>Matched</th>
<th>Mismatched</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clean</td>
<td><img src="audio" alt="Play" /></td>
<td><img src="audio" alt="Play" /></td>
</tr>
<tr>
<td>Noisy</td>
<td><img src="audio" alt="Play" /></td>
<td><img src="audio" alt="Play" /></td>
</tr>
<tr>
<td>RVAE</td>
<td><img src="audio" alt="Play" /></td>
<td><img src="audio" alt="Play" /></td>
</tr>
<tr>
<td>SGMSE+</td>
<td><img src="audio" alt="Play" /></td>
<td><img src="audio" alt="Play" /></td>
</tr>
<tr>
<td>Conv-TasNet</td>
<td><img src="audio" alt="Play" /></td>
<td><img src="audio" alt="Play" /></td>
</tr>
</tbody>
</table>
Interestingly, the same architecture can also be used very well to dereverberate signals!
Conclusions

Our proposed approach:

- Performs on par with state-of-the-art discriminative methods
- Can be applied both to denoising and dereverberation
- Generalizes better under unmatched training conditions


Conclusions
Conclusions

- Deep Neural Networks (DNNs) are very powerful tools for single channel enhancement and source separation
- DNNs allow to estimate spectral phases to allow for high quality speech at low algorithmic latencies
- For multichannel speech enhancement, DNNs can be used to learn joint nonlinear spatial-spectral filters that may outperform the traditional beamformer + postfilter framework
- Diffusion-based generative models are an exciting upcoming field that may increase the robustness in unseen environments