Joint Neural Network Equalizer and Decoder

Weihong Xu[1,2], **Zhiwei Zhong**[2],
Yair Be'ery[3], **Xiaohu You**[2], **Chuan Zhang**[1,2]

[1]Lab of Efficient Architectures for Digital-communication and Signal-processing (LEADS)
[2]National Mobile Communications Research Laboratory, Southeast University, Nanjing, China
[3]School of Electrical Engineering, Tel-Aviv University, Israel

wh.xu@seu.edu.cn

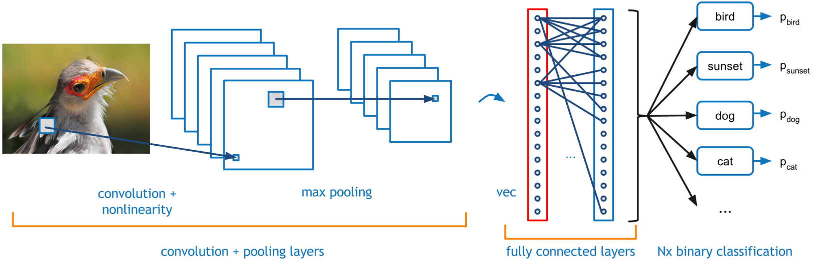
August 29, 2018

Outline

1. Motivation
2. Problem Formulation
3. Proposed CNN Equalizer and DNN Decoder
4. Results and Analysis
5. Conclusion

Outline

Convolutional Neural Networks



- **Convolution**: feature extraction by convolving various filters over input image
- **Fully-connected**: linear transform over input features
- **Pooling and Non-linear**: perform down sampling and non-linear function

Motivation

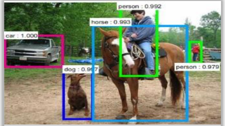
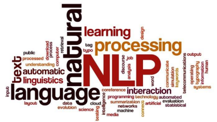


Image Segmentation

Game

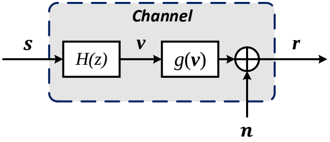


NLP

Outline

1. Motivation
2. Problem Formulation
3. Proposed CNN Equalizer and DNN Decoder
4. Results and Analysis
5. Conclusion

Channel Equalization



- **Inter-symbol Interference (ISI)** The channel with ISI is modeled as a finite impulse response (FIR) filter $\boldsymbol{h}$. The signal with ISI is equivalent to the convolution of channel input with the FIR filter as follows:

$$\boldsymbol{v} = \boldsymbol{s} \otimes \boldsymbol{h}.$$

- **Nonlinear Distortion and Noise** The nonlinearities in the communication system are mainly caused by amplifiers and mixers:

$$r_i = g[v_i] + n_i.$$

Maximum Likelihood Equalizer

▷ **Estimation** Use training sequence $\boldsymbol{s}^\circ$ to estimate channel coefficients $\boldsymbol{h}$ that maximizes likelihood:

$$\hat{\boldsymbol{h}}_{ML} = \arg\max_{\boldsymbol{h}} p(\boldsymbol{r}^\circ | \boldsymbol{s}^\circ, \boldsymbol{h}).$$

▷ **BCJR** Use BCJR algorithm to find codeword that maximizes a posterior probability:

$$p(s_i = s | \boldsymbol{r}, \hat{\boldsymbol{h}}_{ML}), i = 1, 2 \ldots N.$$

▷ **Pros & cons:**

- Good performance for ISI, but bad for nonlinear distortion
- Require accurate channel state information (CSI)
- Complexity: $\mathcal{O}(n^2)$

---

[1] [Olmos, Murillo-Fuentes, and Pérez-Cruz, *TSP* 2010]

SVM for Nonlinear Distortion

- **Nonlinear Distortion** Usually make signal non linearly separable.
- **Support Vector Machine** Perform nonlinear classification for received signal in high-dimensional feature spaces:

$$f(\boldsymbol{x}) = \sum_{i \in S} \alpha_i y_i \Phi(\boldsymbol{x}_i) \cdot \Phi(\boldsymbol{x}) + b.$$

- **Pros & cons:**
  - Improve performance under nonlinear distortion
  - Require proper selection of kernel function $\Phi(\boldsymbol{x})$
  - Complexity: $\mathcal{O}(n)$

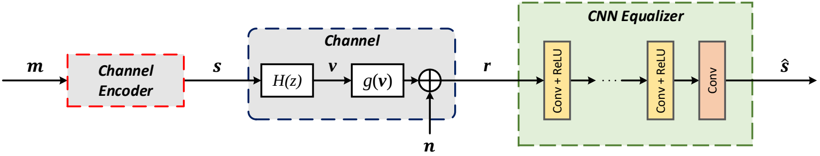
[1] [Sebald and Bucklew, *TSP* 2000]

Problems to Solve

- **A unified framework with high adaptivity**
- **Inaccurate CSI or blind CSI**
- **Channel hard to model**
- **Friendly for hardware design**
- **......**

Outline

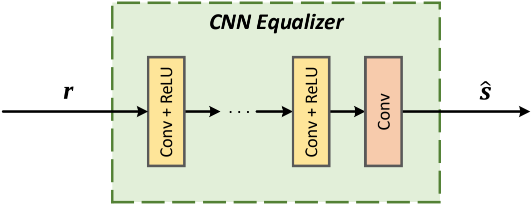
Proposed CNN Equalizer



- Contain trainable filters to cope with various channels.
- End-to-end parameters training to maximize a posteriori:

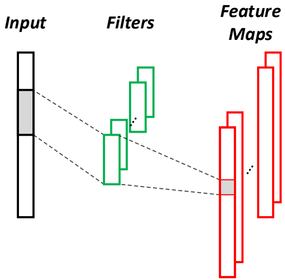
$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} p(\hat{\boldsymbol{s}} = \boldsymbol{s}^{\circ} | \boldsymbol{r}^{\circ}, \boldsymbol{\theta}).$$

Proposed CNN Equalizer



- ▷ Multi-layer fully convolutional network with ReLU activation.
- ▷ The last layer only contains convolution operation.

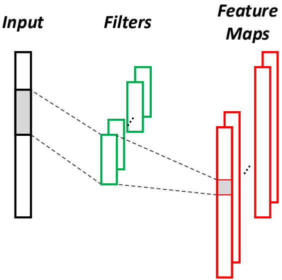
Proposed CNN Equalizer



- ▷ Fully convolutional neural network with 1-D convolution:

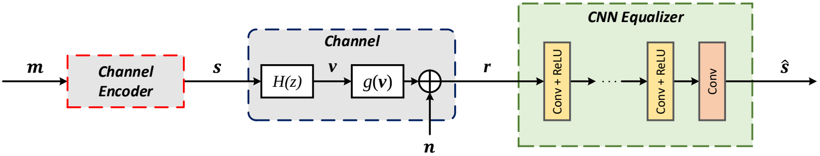
$$\mathbf{y}_{i,j} = \sigma(\sum_{c=1}^{C}\sum_{k=1}^{K}\mathbf{W}_{i,c,k}\mathbf{x}_{c,k+j} + b_i).$$

Proposed CNN Equalizer



- Use 1×3 sized filters and padding $=1$.
- To keep the signal dimension constant.
- Avoid information loss.

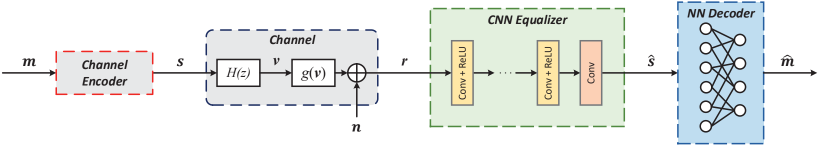
Proposed CNN Equalizer



- **Mean squared error (MSE)** loss function:

$$\mathcal{L}(\hat{\boldsymbol{s}}, \boldsymbol{s}) = \frac{1}{N} \sum_i |\hat{s}_i - s_i|^2.$$

Proposed DNN Decoder

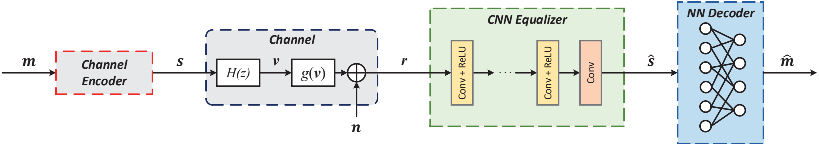


- ▷ A deep neural network (DNN) polar decoder:

$$\mathbf{y} = \sigma(\mathbf{W}\mathbf{x} + \mathbf{b}).$$

- ▷ DNN decoder with structure {16, 128, 64, 32, 8}.

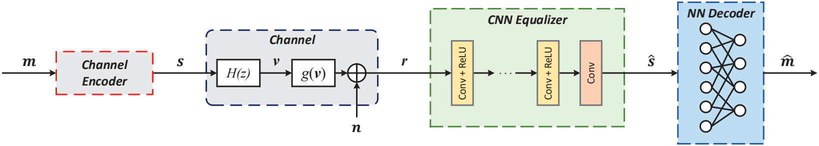
Proposed DNN Decoder



▷ **Binary cross entropy (BCE)** loss function:

$$\mathcal{L}(\hat{\boldsymbol{m}}, \boldsymbol{m}) = \frac{1}{N} \sum_{i} \hat{m}_i \log(m_i) + (1 - \hat{m}_i) \log(1 - m_i).$$

Joint Training of CNN and DNN



- ▷ The networks can be trained separately or jointly.
- ▷ Joint training is easier to optimize.
- ▷ Total loss of equalization and decoding:

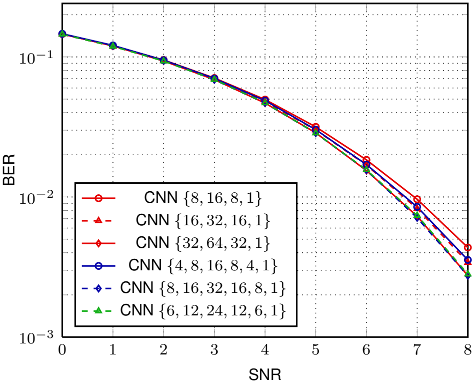
$$\mathcal{L}_{total} = \mathcal{L}(\hat{\boldsymbol{s}}, \boldsymbol{s}) + \mathcal{L}(\hat{\boldsymbol{m}}, \boldsymbol{m}).$$

Outline

Experiment Details

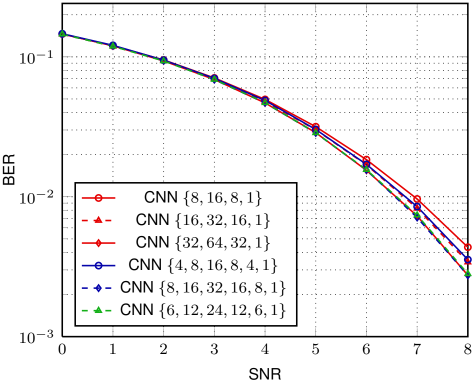
- **CNN structure:** 6 layers
- **Learning rate** $= 0.001$
- **Random codewords from SNR 0 dB to 11 dB**
- **Mini-batch size: 240 (20 per SNR)**
- **Weights initialization:** $\mathcal{N} \sim (\mu = 0, \sigma = 1)$
- **Polar code**: (16,8)

Select the Optimal Network Structure



- ▷ Deeper networks $\longrightarrow$ Better performance
- ▷ More filters $\longrightarrow$ Better performance

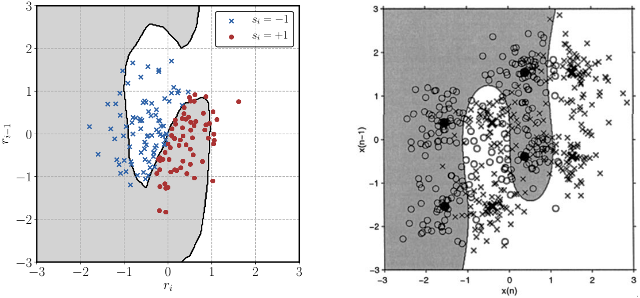
Select the Optimal Network Structure



- CNN {6, 12, 24, 12, 6, 1} (containing 2257 weights) is close to CNN {32, 64, 32, 1} (containing 12609 weights).

Decision Boundaries

- ▷ Test under $\boldsymbol{h} = [1, 0.5]$ and $g(v) = v - 0.9v^3$.
- ▷ CNN has nonlinear decision boundaries similar to SVM.



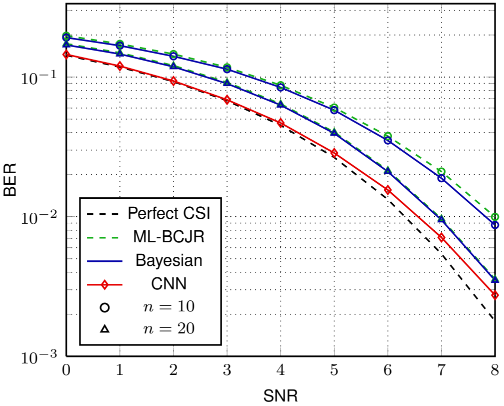
(a) CNN (b) SVM

[1][Sebald and Bucklew, *TSP* 2000]

Results on Linear Channel with ISI

- Channel coefficients:

$$H(z) = 0.3482 + 8704z^{-1} + 0.3482z^{-2}.$$

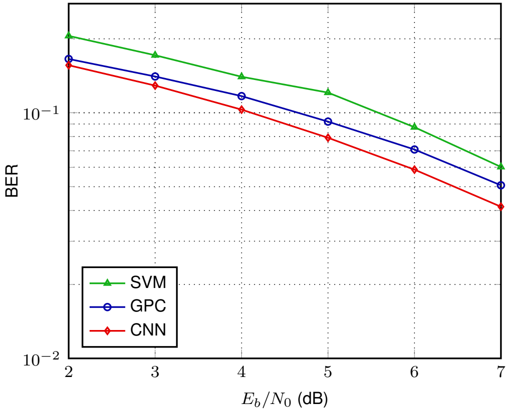


[1][Salamanca, Murillo-Fuentes, and Pérez-Cruz, *ISIT* 2010]

Results on Nonlinear Channel with ISI

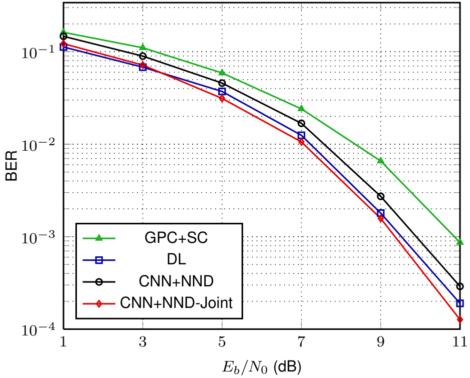
- Nonlinear distortion:

$$|g(v)| = |v| + 0.2|v|^2 - 0.1|v|^3 + 0.5\cos(\pi|v|).$$



[1] [Olmos, Murillo-Fuentes, and Pérez-Cruz, *TSP* 2010]

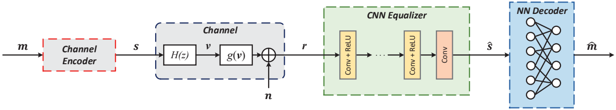
Results on Joint DNN-CNN Detector

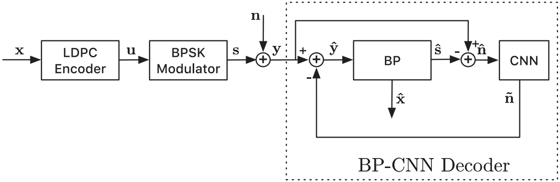


- Outperforms GPC+SC.
- Only requires about $1/3$ parameters compared with DL method.

[1] [Ye and Li, *VTC-Fall* 2017]

Analysis





- Proposed design: CNN receives channel output.
- BP-CNN decoder in [5]: CNN receives output of BP decoder.

[1] [Liang, Shen, and Wu, *JSTSP* 2018]

Analysis

- Robust to different channel conditions.
- Computation complexity is about $\mathcal{O}(n)$ due to the 1-D convolution.
- Support various sequence lengths without retraining.

Outline

Conclusion

- **A channel equalizer based on CNN**
  - Soft-in soft-out equalizer
  - Feasible for long code
- **Near optimal performance under various channels**
  - No need for accurate CSI
  - Linear complexity: $\mathcal{O}(n)$
- **Friendly to hardware design**
  - Regular dataflow
  - Advanced CNN hardware architectures

Future Work

- **Adopt depth-wise separable convolution**
  - Prevent information loss
  - Reduce computation complexity
- **High order modulation**
  - Neural networks on complex domain

Reference

1. P. M. Olmos, J. J. Murillo-Fuentes, and F. Pérez-Cruz, "Joint nonlinear channel equalization and soft ldpc decoding with gaussian processes," *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1183–1192, 2010
2. L. Salamanca, J. J. Murillo-Fuentes, and F. Pérez-Cruz, "Channel decoding with a bayesian equalizer," in *IEEE International Symposium on Information Theory Proceedings (ISIT)*, 2010, pp. 1998–2002
3. H. Ye and G. Y. Li, "Initial results on deep learning for joint channel equalization and decoding," in *IEEE Vehicular Technology Conference (VTC-Fall)*, 2017, pp. 1–5
4. T. Gruber, S. Cammerer, J. Hoydis, *et al.*, "On deep learning-based channel decoding," in *Annual Conference on Information Sciences and Systems (CISS)*, IEEE, 2017, pp. 1–6
5. D. J. Sebald and J. A. Bucklew, "Support vector machine techniques for nonlinear equalization," *IEEE Transactions on Signal Processing*, vol. 48, no. 11, pp. 3217–3226, 2000
6. F. Liang, C. Shen, and F. Wu, "An iterative bp-cnn architecture for channel decoding," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 144–159, 2018

Thanks for Your Attention!

Q & A