2024 IEEE International Conference on Future Machine Learning and Data Science (FMLDS) | 979-8-3503-9121-3/24/$31.00 ©2024 IEEE | DOI: 10.1109/FMLDS63805.2024.00066

Image Separation Using Transformer Attention Models

Aditya Ranganath
Lawrence Livermore National Laboratory
Livermore, USA
ranganath2@llnl.gov

Jocelyn Ornelas Muñoz
Applied Mathematics
University of California, Merced
Merced, USA
jornelasmunoz@ucmerced.edu

Robert Smith
School of Engineering
University of California, Merced
Merced, USA
rsmith49@ucmerced.edu

Mukesh Singhal
School of Engineering
University of California, Merced
Merced, USA
msinghal@ucmerced.edu

Roummel Marcia
Applied Mathematics
University of California, Merced
Merced, USA
rmarcia@ucmerced.edu

***Abstract*—Signal recovery often involves separating and realizing multiple superimposed signals at once. Separating multiple images that have been superimposed is a challenging signal recovery problem. This situation arises when a detector, such as a microphone, receives multiple signals simultaneously. In order to recover the original signals, a signal separator needs to be applied. In this paper, we will explore machine learning techniques for separating such signals. In particular, we investigate two approaches: an autoencoder approach and a transformer-based approach, and test their accuracy in recovering two separate images from noisy low-resolution superimposed measurements.**

***Index Terms*—Image separation, machine learning, deep learning, denoising, transformers**

I. Introduction

Image separation is a common signal separation problem in the domain of signal processing. Commonly referred to as 'blind source separation' (BSS), the problem involves separation of source signals with very little information about the sources or the multiplexing operation [1], [2].

Much of the early literature focuses on separation of temporal signals, such as audio [3], [4] or video [5]. However, BSS has gained momentum in the field of images and tensors, which may have no temporal component whatsoever (see [6]). There is also literature which uses deep learning for blind source separation [7]. In contrast, for practical applications in digital imaging, noises can be caused by sudden change in light intensity, increase in temperature of the imaging apparatus or electrical fluctuations during transmission of the signal. Typically this type of noise is modeled as additive white Gaussian noise (AWGN). In the event that the imaging apparatus records the images with low resolution, the images may be compressed as well. Thus, in addition to BSS, the image noise and compression need to be tackled.

In this paper, we explore two deep learning strategies to address all of these issues simultaneously. The paper is organized as follows: In Sec. II, we discuss the blind source problem formulation, in Sec. III, we discuss the proposed approaches for separting the signals, in Sec. IV, we describe the numerical experiments of the proposed apporaches and in Sec. V and Sec. VI, we discuss the results and conclude the paper respectively.

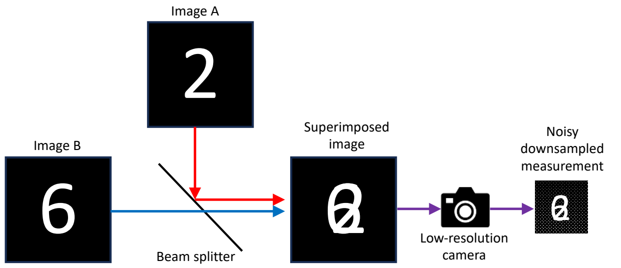


Fig. 1. Schematic of the imaging system. Two images (A and B) are superimposed using a beam splitter observed at the detector of a low-resolution camera, resulting in a downsampled measurement with additive white Gaussian noise.

II. Problem Formulation

The blind source multiplexed problem can be formulated as

$$\mathbf{y} = \mathbf{D}(\mathbf{z}) + \mathbf{g}, \tag{1}$$

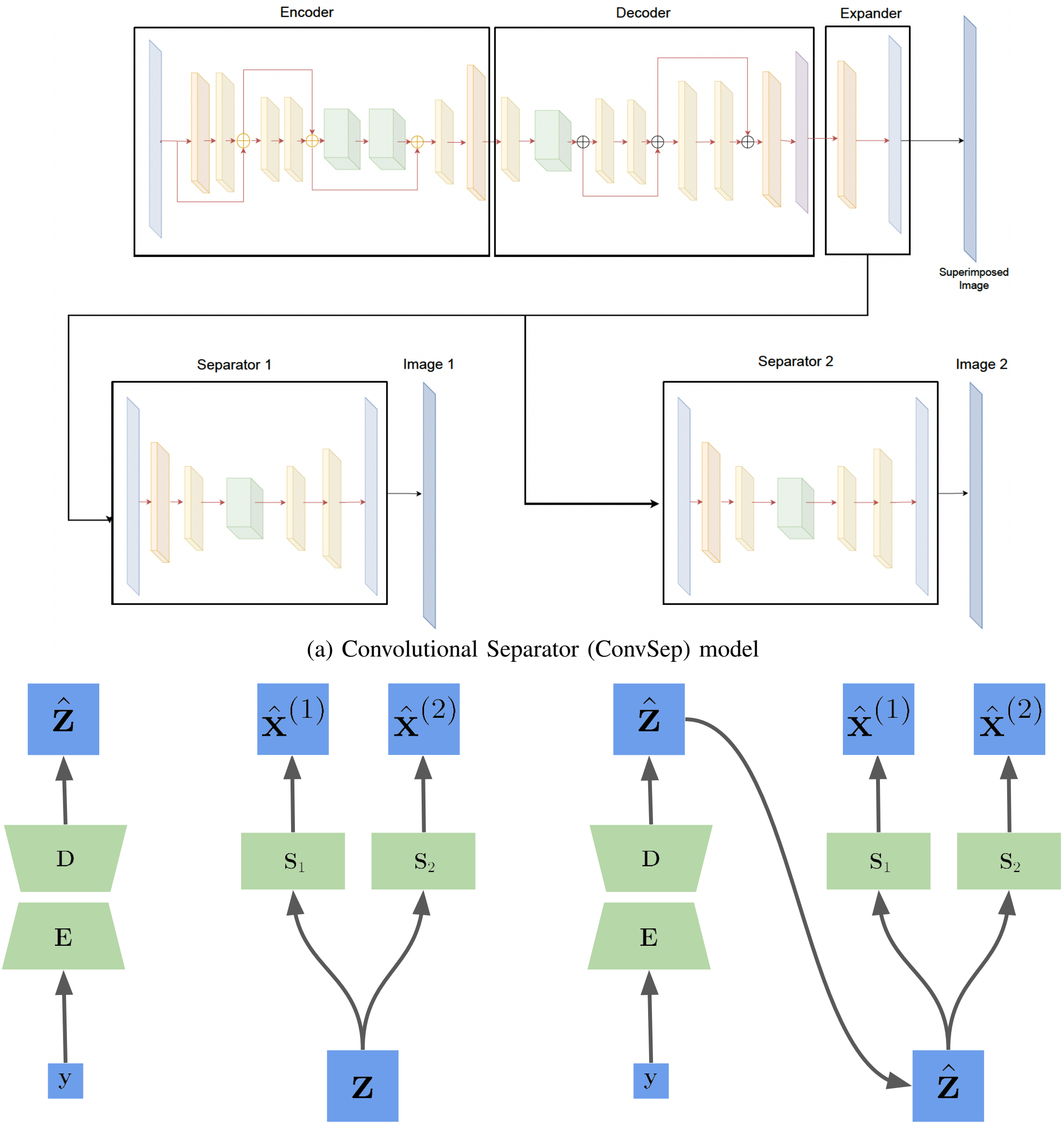
where $\mathbf{D}(\mathbf{z})$ is the downsampling operator and

$$\mathbf{z} = \mathbf{x}^{(1)} + \mathbf{x}^{(2)},$$

i.e., $\mathbf{z}$ is the resulting image of superimposing two images $\mathbf{x}^{(1)}$ an $\mathbf{x}^{(2)}$. The vector $\mathbf{g} \sim \mathcal{N}(0, \sigma^2)$ is additive white Gaussian noise with zero mean and variance σ^2. These operations describe the linear model of observing noisy low-resolution images that are superimposed at the detector stage (see Fig. 1).

Related work: In [8], the authors use a stacked autoencoder with fully connected layers. However, the size of the network can get prohibitively expensive for larger images. Also,

979-8-3503-9121-3/24/$31.00 ©2024 IEEE
DOI 10.1109/FMLDS63805.2024.00066



(a) Convolutional Separator (ConvSep) model

(b) ConvSep training procedure

(c) ConvSep testing procedure

Fig. 2. Illustration of the Model I: Convolutional Separator (ConvSep) approach. (a) ConvSep contains an encoder, a decoder, an expander, and two separators. The colored box represents the output from a convolutional operator. (b) During training, the superimposed and clean images, $\mathbf{Z}$, are available to the ConvSep network to perform the separating operation. (c) During testing, the superimposed, clean images are not available to the network. Instead the output, $\hat{\mathbf{Z}}$, from the decoder, $\mathbf{D}$, is fed to the separators, $\mathbf{S}_1$ and $\mathbf{S}_2$.

the authors focus on weighted multiplexing problem without denoising and downsampling. In [9], [10], the authors use a convolutional neural network for denoising images. They use two approaches to denoise the images - an autoencoder with convolutional layers and a recurrent neural network with convolutional layers as hidden units. This was able to tackle the problem of reducing the footprint of the network by replacing the linear layers with convolutional layers. In addition, the authors were able to realise the noise in the images as a temporal component. Our first method is based on this approach.

Attention-based transformers have gained much momentum in the last few years. The concept of transformers was introduced in [11] for natural language processing (NLP) [12], [13]. With increased improvement in NLP applications, the use of transformers was pervasive in many different fields, such as image denoising [14], [15], protein structure prediction [16], and sentiment analysis [17]. However, this approach has not been applied in an image separation regime.

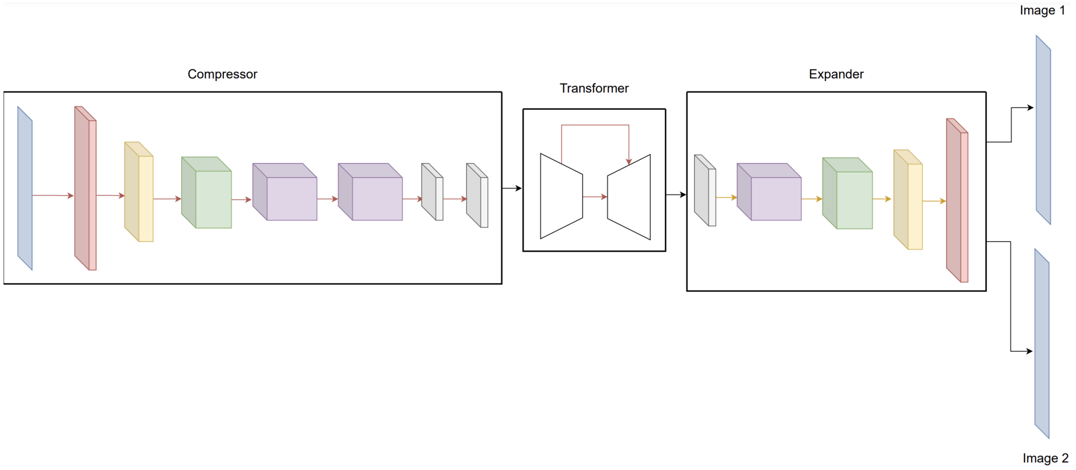


Fig. 3. Illustration of the Model II: Limited-informed Generative Trasnformer (LiGT) approach. This transformer-based model is composed of a compressor, a transformer and an expander. Each colored box represents the output from a convolutional operator.

III. Proposed approach

In this section, we describe the two approaches, the corresponding loss functions, and their respective datasets.

Model I (ConvSep): In [8], the authors use a fully-connected stacked autoencoder architecture to denoise the images. This significantly increases the model footprint and the time taken to train the models. To tackle this problem, we propose the **Convolutional Separator (ConvSep)** model. The novelty of the model lies in its convolution operation; convolution operators occupies a smaller memory footprint than a fully connected autoencoder, resulting in a much faster to training response. In this approach, we take the compressed, noisy realization, and expand the dimensions using an encoder-decoder operation. Then we use two separators that separate the image into their two parent images. The ConvSep model is illustrated in Fig. 2(a).

During the training operation, the superimposed image is available to perform the image expansion step. For more details on how this model operates, please refer Fig. 2(b).

Model II (LiGT): The second proposed method, which we call Limited-informed Generative Transformer (LiGT), is a transformer based model which expands, cleans and separates the mutliplexed signal at once. Fig. 3 shows the disambiguation operation for LiGT. The features of the image are extracted using a Compressor. This feature is then fed to the Transformer encoder. The transformer decoder extracts the cross-attention between the features and the output encoder. The output of the transformer decoder is fed to the Expander, which expands the dimensions of the output from the transformer decoder into the two parent images.

The novelty of the method lies in using the transformer with a limited data setting. To the knowledge of the authors, this is the first time a transformer based approach has been used to denoise and separate images. Unlike the ConvSep model, the superimposed, clean and upsampled image is not available to the model, thus lacking information. In addition, the memory footprint of the model is also smaller in comparison to the ConvSep model.

Loss function: We optimize the network parameters using HuberLoss, which is defined as

$$\mathcal{L}_\delta(\hat{\mathbf{x}},\mathbf{x}) = \begin{cases} \frac{1}{2}\|\hat{\mathbf{x}}-\mathbf{x}\|_1^2 & \text{if } \|\hat{\mathbf{x}}-\mathbf{x}\|_1 < \delta \\ \delta(\|\hat{\mathbf{x}}-\mathbf{x}\|_1 - \frac{1}{2}\delta) & \text{otherwise} \end{cases}, \quad (2)$$

where $\delta \in \mathbb{R}$ makes the loss function 'differentiable' at prohibitively small values of the absolute difference, $\hat{\mathbf{x}} \in \mathbb{R}^{n\times n}$ is the reconstructed realization and $\mathbf{x} \in \mathbb{R}^{n\times n}$ is the true image. We choose a value of $\delta = 1$ for our experiments.

The loss function for Model I: ConvSep is given by

$$\mathcal{L}_{\text{ConvSep}} = \mathcal{L}(\hat{\mathbf{z}},\mathbf{z}) + \mathcal{L}(\hat{\mathbf{x}}_1,\mathbf{x}_1) + \mathcal{L}(\hat{\mathbf{x}}_2,\mathbf{x}_2), \quad (3)$$

where $\mathcal{L}(\mathbf{z},\hat{\mathbf{z}})$ is the loss between the denoised and upsampled superimposed ground truth and reconstructed images, $\mathcal{L}(\hat{\mathbf{x}},\mathbf{x})$ is the loss between the reconstructed image $\hat{\mathbf{x}}$ and the ground truth image $\mathbf{x}$. This model has a total of 2,846,224 parameters.

The loss function for Model II: LiGT is given by

$$\mathcal{L}_{\text{LiGT}} = \mathcal{L}(\hat{\mathbf{x}}_1,\mathbf{x}_1) + \mathcal{L}(\hat{\mathbf{x}}_2,\mathbf{x}_2). \quad (4)$$

This model has a total of 2,301,865 parameters.

Dataset: We use the MNIST dataset [18] in our experiments. To generate our data, we randomly choose two images $\mathbf{x}_i^{(1)}, \mathbf{x}_i^{(2)} \in \mathbb{R}^{28\times 28}$ from the MNIST data and superimpose them to obtain $\mathbf{z}_i \in \mathbb{R}^{28\times 28}$, which is then downsampled by a factor of 2 and to which AWGN is added to yield the noisy low-dimensional superimposed images $\mathbf{y}_i \in \mathbb{R}^{14\times 14}$. For Method I, the dataset is given by $\mathcal{D}_1 = \{\mathbf{x}_i^{(1)}, \mathbf{x}_i^{(2)}, \mathbf{z}_i, \mathbf{y}_i\}_{i=1}^N$. For Method II, the noisy, downsampled and superimposed observation is directly mapped to the two clean realizations and the intermediary data $\mathbf{z}_i$ is not used. The dataset is thus given by $\mathcal{D}_2 = \{\mathbf{x}_i^{(1)}, \mathbf{x}_i^{(2)}, \mathbf{y}_i\}_{i=1}^N$.

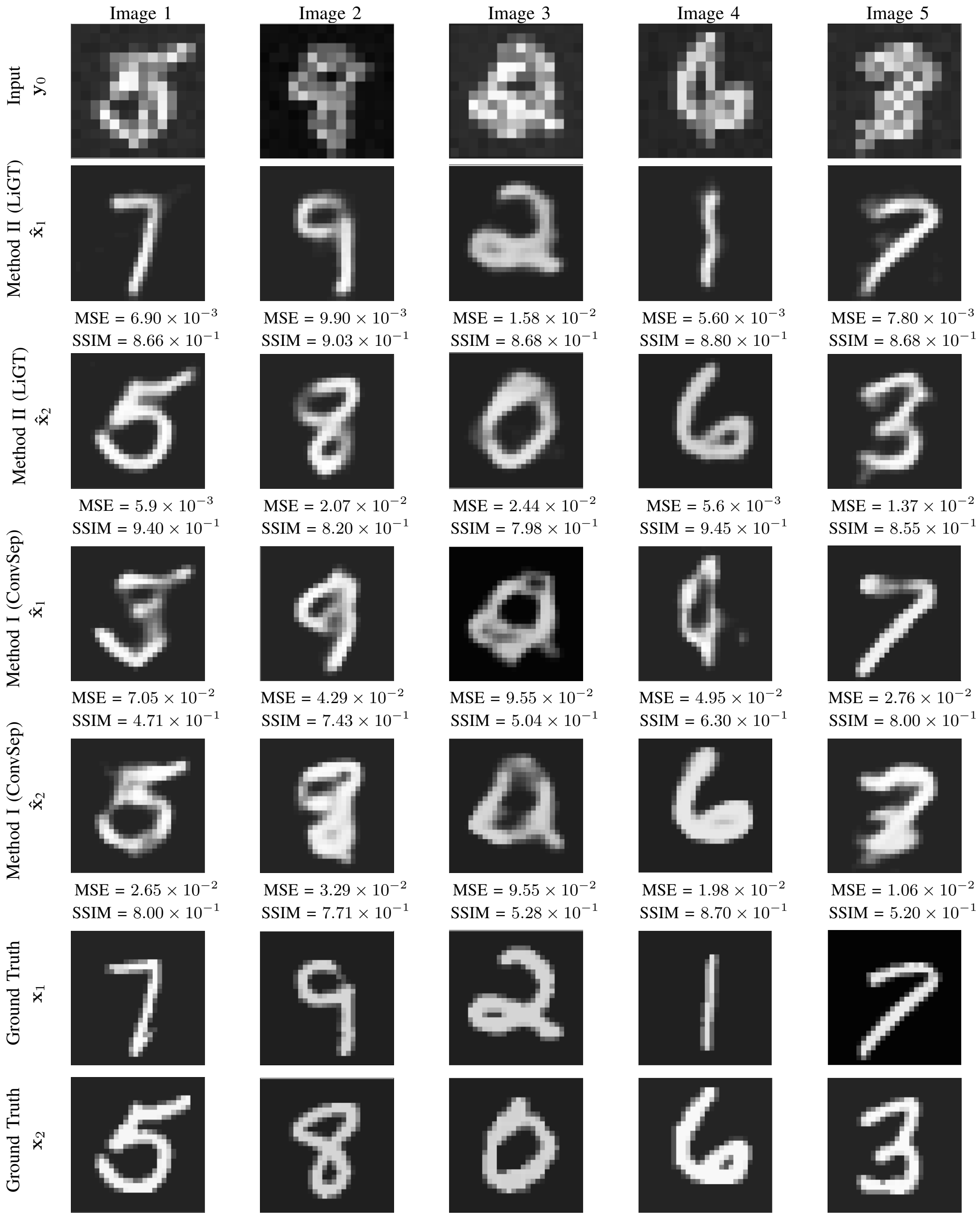
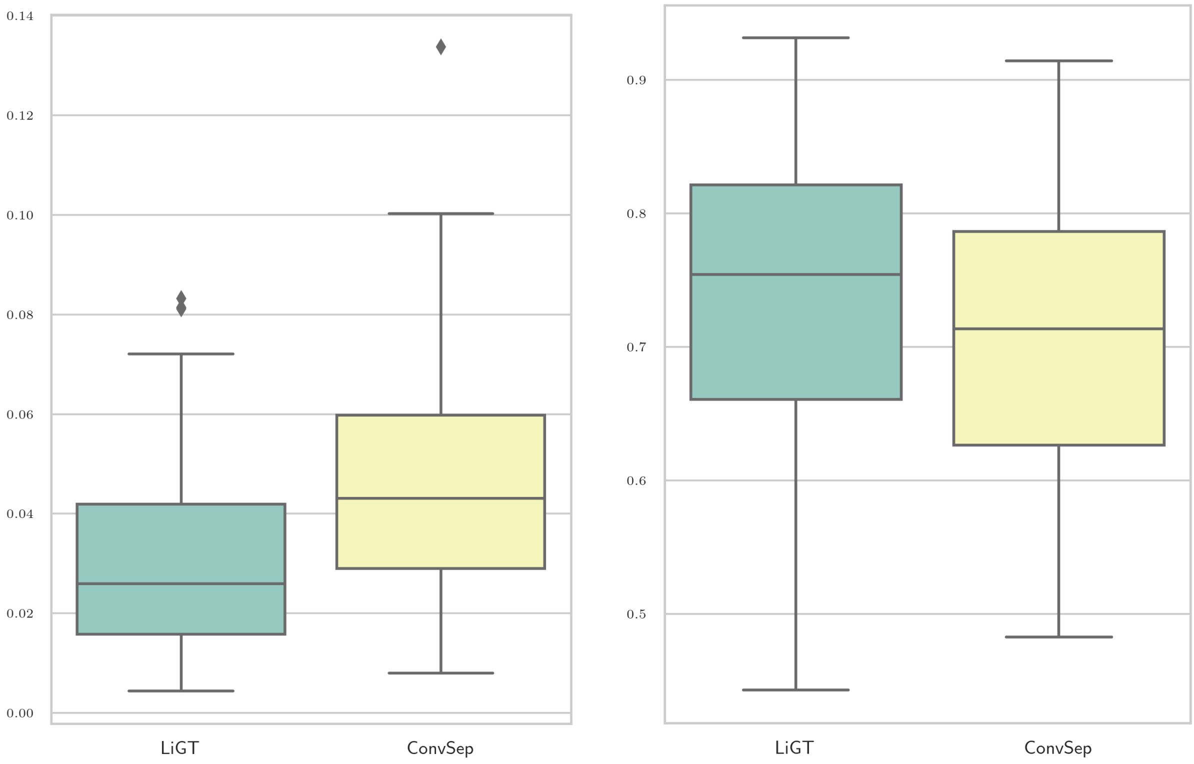


Fig. 4. Numerical experiments on 5 images from the MNIST dataset. Row 1: Noisy input images $\mathbf{y}$. Rows 2 and 3: Final reconstructions $\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2$ using Method II (LiGT). Rows 4 and 5: Final reconstructions $\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2$ using Method I (ConvSep). Rows 6 and 7: Ground truth images $\mathbf{x}_1$, $\mathbf{x}_2$. MSE and SSIM values for both Methods I (ConvSep) and II (LiGT) are presented for each image.



(a) MSE

(b) SSIM

Fig. 5. Box plots of the collective results for Method II: LiGT (in green) and Method I: ConvSep (in yellow). (a) Mean-squared error (MSE). (b) Structural similarity index metric (SSIM).

IV. Experiments

In this section we describe the testbed and the training procedures for both the models. All the architectures were implemented using PyTorch [19]. Training and testing were performed using two NVIDIA 1080 Ti GPUs. The networks were trained using the Adam optimizer [20].

Training: During training, the superimposed, compressed and noisy images are fed to both the models. For the LiGT model, these images are directly mapped to the clean and separated images. For the ConvSep model, the downsampled and noisy superimposed observation is fed to the encoder $\mathbf{E}$ for the $\mathbf{D}$ to yield the clean, upsampled superimposed construction $\hat{\mathbf{z}}$. The clean, upsampled superimposed image $\mathbf{z}$ is then fed to the separators $\mathbf{S}_1$ and $\mathbf{S}_2$ to yield $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_2$.

Testing: During testing, the operation of LiGT matches the training procedure. However, for the ConvSep model, the upsampled superimposed construction $\hat{\mathbf{z}}$ is directly fed to separators $\mathbf{S}_1$ and $\mathbf{S}_2$ (see Fig. 2(c) for illustration).

V. Results

In this section, we present the results from the two approaches. Fig. 4 shows the results for both the approaches. The first row shows the superimposed images, downsampled with added noise. The second and third row show the separated images using the LiGT model, the fourth and fifth row show the images separated using the ConvSep model. The last two images show the ground truth images. We can notice that the MSE for the LiGT model is much lower than the ConvSep model. It can also be noticed that the LiGT model was able to improve of the structural integrity of the image better than the ConvSep model. Fig. 5 presents the overall MSE and SSIM results. The average MSE loss for the ConvSep model was 4.66×10^{-2}, and the average MSE loss for the LiGT model was 3.00×10^{-2}. The SSIM value for ConvSep model was 0.75 and the SSIM value for LiGT model was 0.72.

VI. Conclusion

In this paper, we presented two approaches for image disambiguation. The first approach (ConvSep) uses an RNN-inspired convolutional neural network to denoise and upsample in one stage and disambiguate the images in another stage. The second approach (LiGT) is a transformer-based model which denoises, upsamples and disambiguates the image simultaneously. Experiments and results show that the transformer-based model was able to outperform the RNN inspired approach with a smaller model footprint.

VII. Acknowledgement

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. R. Marcia's research is partially supported by NSF Grant DMS 1840265.

References

[1] X.-R. Cao and R.-w. Liu, "General approach to blind source separation," *IEEE Transactions on signal Processing*, vol. 44, no. 3, pp. 562–571, 1996.

[2] G. R. Naik, W. Wang, *et al.*, "Blind source separation," *Berlin: Springer*, vol. 10, pp. 978–3, 2014.

[3] K. Rahbar and J. P. Reilly, "A frequency domain method for blind source separation of convolutive audio mixtures," *IEEE Transactions on speech and audio processing*, vol. 13, no. 5, pp. 832–844, 2005.

[4] A. Liutkus, J.-L. Durrieu, L. Daudet, and G. Richard, "An overview of informed audio source separation," in *2013 14th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS)*, pp. 1–4, IEEE, 2013.

[5] R. F. Marcia, C. Kim, C. Eldeniz, J. Kim, D. J. Brady, and R. M. Willett, "Superimposed video disambiguation for increased field of view," *Opt. Express*, vol. 16, no. 21, pp. 16352–16363, 2008.

[6] I. Meganem, Y. Deville, S. Hosseini, P. Deliot, and X. Briottet, "Linear-quadratic blind source separation using nmf to unmix urban hyperspectral images," *IEEE Transactions on Signal Processing*, vol. 62, no. 7, pp. 1822–1833, 2014.

[7] X. Yu, D. Hu, and J. Xu, *Blind source separation: theory and applications*. John Wiley & Sons, 2013.

[8] O. DeGuchy, A. Ho, and R. F. Marcia, "Image disambiguation with deep neural networks," in *Applications of Machine Learning*, vol. 11139, pp. 68–74, SPIE, 2019.

[9] A. Ranganath, O. DeGuchy, M. Singhal, and R. F. Marcia, "Multi-stage gaussian noise reduction with recurrent neural networks," in *2021 55th Asilomar Conference on Signals, Systems, and Computers*, pp. 135–139, 2021.

[10] A. Ranganath, O. DeGuchy, F. Santiago, M. Singhal, and R. Marcia, "Recurrent nerual imaging: An evolutionary approach for mixed possion-gaussian image denoising," in *2022 21st IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 484–489, 2022.

[11] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[12] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[13] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[14] D. Zhang and F. Zhou, "Self-supervised image denoising for real-world images with context-aware transformer," *IEEE Access*, vol. 11, pp. 14340–14349, 2023.

[15] S. Pan, T. Wang, R. L. Qiu, M. Axente, C.-W. Chang, J. Peng, A. B. Patel, J. Shelton, S. A. Patel, J. Roper, *et al.*, "2d medical image synthesis using transformer-based denoising diffusion probabilistic model," *Physics in Medicine & Biology*, vol. 68, no. 10, p. 105004, 2023.

[16] R. M. Rao, J. Liu, R. Verkuil, J. Meier, J. Canny, P. Abbeel, T. Sercu, and A. Rives, "Msa transformer," in *International Conference on Machine Learning*, pp. 8844–8856, PMLR, 2021.

[17] U. Naseem, I. Razzak, K. Musial, and M. Imran, "Transformer based deep intelligent contextual embedding for twitter sentiment analysis," *Future Generation Computer Systems*, vol. 113, pp. 58–69, 2020.

[18] Y. LeCun, C. Cortes, C. Burges, *et al.*, "Mnist handwritten digit database," 2010.

[19] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*, pp. 8024–8035, Curran Associates, Inc., 2019.

[20] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.