浙江大学

# 本科生毕业论文（设计）

# 文献综述和开题报告


姓名与学号　曾鹏熹 3180105152

指导教师　黄崇文

年级与专业　2018级信息工程

所在学院　信息与电子工程学院

一、题目：智启无线通信：基于 AI 的信道估计技术研究

二、指导教师对文献综述、开题报告、外文翻译的具体要求

文献综述和开题报告的要求：

**1**．查阅国内外相关文献资料 **8-10** 篇以上（其中外文文献不少于 **3-5** 篇）；完成至少一篇 **3000** 字以上外文翻译。

**2**．完成文献综述（**3000** 字以上）和开题报告（**3500** 字以上）。

对文献综述的具体要求是：

（1）了解解调参考信号 **DMRS**（**Demodulation Reference Signal**）的特点，了解 **DMRS** 信号应用以及国内外发展现状。

（2） 阐述已有的 **DMRS** 信道估计算法以及其国内外研究现状。

（3） 比较已有的信道估计算法，阐述引入 **AI** 进行信道估计的应用及其研究意义。

（4） 做好毕业论文相关知识的准备工作。

对开题报告的具体要求是：

（1） 分析当前用于 **DMRS** 信道估计算法的种类与特点。

（2） 给出基于 **AI** 的 **DMRS** 信道估计大致设计方案与思路。

（3）对论文要完成的工作有清晰的认识，给出具体的研究方案，预见可能遇到的问题和困难。

（4） 提出论文的研究成果或预期达到的目标。

（5） 安排好论文的进度，对于每个阶段的工作有详细的规划。

指导教师（签名）_____

2021 年 11 月 6 日

# 目录

# 一、文献综述

## 1 简介

自二十世纪之交以来，无线通信技术以惊人的速度发展。数据传输的速度也大大加快了，无限通信网络的带宽也随之迅速提高。因此，通信系统的质量已成为当今世界的关键因素。对于任何通信系统，信道估计都是至关重要的，因为信道估计的准确性会影响整个系统的质量。

在过去几年中，针对不同类型通信系统的各种传统信道估计算法进行了大量研究[1]。借助于仿真，根据各种参数（如信噪比（SNR）、误码率（BER）、均方误差（MSE）等）比较各种信道估计方案，以确定哪种方案最适合于特定类型的通信系统。对于无线正交频分复用（OFDM）系统，提出了最小均方（MMSE）估计器和最小二乘（LS）估计器[2]，还在时变色散信道中的微蜂窝 OFDM 上行链路中提出了基于最大似然的信道估计[3]和大型天线系统中的数据辅助信道估计方案[4]。而使用深度学习算法是信道估计系统的一个新趋势。深度学习是学习样本数据的内在规律和表示层次，这些学习过程中获得的信息对诸如文字，图像和声音等数据的解释有很大的帮助。它的最终目标是让机器能够像人一样具有分析学习能力，能够识别文字、图像和声音等数据。 深度学习是一个复杂的机器学习算法，在语音和图像识别方面取得的效果，远远超过先前相关技术。与传统方法相比，基于深度学习的技术的优点之一是其鲁棒性。尤其是当数据量较大时，深度学习具有优于传统方法的描述数据特征的良好表征能力。因此，在我们二十一世纪的无线通信系统中，由于数据速率和带宽日益增长，深度学习算法成为信号识别和信道估计领域的一种具有良好潜力的工具。在信道估计领域，深度学习已经崭露头角，但仍然存在一些问题，

接下来我们将简单介绍信道估计的理论；用于信道估计的不同的深度学习的架构以及相关挑战；最后提出该问题的研究展望。

## 2 信道估计理论

在实际的通信系统中，任何信号在传输过程中均会受到由于不理想信道带来的污染，多种噪声会加在信号本身，给接收端的解调以及检测等工作带来很大的

阻碍。为了能够提高通信系统的效率，我们需要尽可能消除不理想信道导致的信号扭曲，而这就需要我们对信道的特征进行描述，也就是信道估计。通常信道估计是通过比较已知信号在收发机两端的不同来获得大致的信道矩阵。整体过程为：首先在发射机端，我们发送一系列已知信号，即导频信号，这些导频信号通常经过一定的设计使其便于在接收端检测得到；随后这些信号通过信道被噪声扭曲；最后我们在接收机端收到相关信号，通过比较收发信号的差异，我们得到其相关关系，从而得到对信道的估计。

# 3 基于深度学习的信道估计算法概述

在机器学习中，根据训练方式大致可以分为有监督学习、无监督学习与强化学习。其中有监督学习被广泛应用于图像处理领域。在通信领域，对于基于深度学习的信达估计算法，我们不需要进行分类，而是需要获得数据的内在关系，拟合一定的算法，故一般视为无监督学习。

多数论文的深度学习架构基于经典深度学习结构，使用线性层或卷积层的连接来进行信道估计。在论文[5]中，作者提出了一种基于深度学习的信道估计算法，主要针对毫米波大规模 MIMO 系统。在该应用场景中，MIMO 系统的天线阵列数量巨大，而射频链路的数量却相对来说非常小，从而使得由收到的信号估计信道这项任务变得具有很大的挑战性。该论文针对该应用场景，提出了基于经典深度学习架构的 LDAMP（learned denoising-based approximate message passing）网络，其中每一层均为相同的结构，通过多层连接，逐步学习得到信道估计。其中的去噪模块，论文使用了计算机视觉领域（CV）的相关工作 DnCNN（denoising convolutional neural network）[6]来实现。DnCNN 的结构与经典卷积网络 VGG 类似，通过多层卷积层连接，但是最后其学习得到的是残差图像，即为噪声图像。作者通过仿真结果证明了 LDAMP 网络的优越性能，同时比较了 LDAMP 网络、SD 算法、SCAMP 算法以及 DAMP 算法，得出去噪模块的使用使得 LDAMP 网络和 DAMP 算法获得了性能上的提升。

在[7]中，研究团队针对毫米波大规模 MIMO 系统设计了基于与[5]相同的应用场景。论文逐步提出了三种架构，分别为空间-频率卷积网络（SF-CNN）、空间-频率-时间卷积网络（SFT-CNN）和空间导频减少卷积网络（SPR-CNN），其

中 SFT-CNN 比 SF-CNN 能更好地提取到信道时域上的相关性，而 SPR-CNN 能够在一定程度上减少导频的使用，节省频谱资源。数值结果表明，SFT-CNN 能够取得与理想 MMSE 相近的结果，同时大大节省计算量。而 SPR-CNN 能够在减少导频的情况下仍然有与 SF-CNN 相近的性能。

又如[8]，在文中作者提出了针对 MIMO 多用户系统基于深度学习的信道估计算法以及导频设计。他们使用一个两层的神经网络设计导频，以及一个 DNN 用于估计信道。该文提出导频长度可以压缩，虽然会失去正交性，但是利用深度学习的非线性拟合能力可以在一定程度上解决这个问题。模拟实验的数值结果证明在此情形下，提出的深度学习算法取得了远超过 LMMSE 的性能。

部分论文尝试深度学习与经典算法相结合，在论文[9]中，作者针对在双选择性衰落信道情形下，难以精确建模的问题，设计了最小二乘（LS）与 DNN 混合的深度学习架构，将 LS 估计得到的信道信息作为特征输入，以已知全信号获得对于信道的估计。获得了逼近线性最小均方误差（LMMSE）的性能。

另外多篇论文将信道矩阵视为二维图像，尝试通过 CV 的方法重建信道矩阵。如在[10]中，一种基于深度学习的信道估计算法被提出，名为 ChannelNet。该论文快衰落 OFDM 信道的时频响应视为一张 2D 图像，其目的是利用导频作为已知信号，获得信道响应的未知值。该论文提出了一种基于深度学习图像处理的方法。导频信号的响应被视为视频响应矩阵的部分采样，通过图片超分辨率重建（Super Resolution, SR）与图像恢复（Image Restoration， IR）来获得完整的时频响应矩阵。首先，图片超分辨率重建模块估计未知位置的时频响应，随后通过图像恢复模块来消除噪声的影响。该论文采用了基于深度卷积网络的深度图像算法分别实现超分辨率模块和图像恢复模块，分别为 SRCNN[11]和 DnCNN[6]。其中 SRCNN 首先通过线性插值大致恢复至原始图像尺寸，随后通过三层卷积网络进行更加精确的重建。论文通过数值实验发现，在信噪比低于 20 dB 时，该算法取得了逼近理想 MMSE 的优良性能，而在高于 23 dB 时，性能有所退化，需要设计、训练新的网络。

此外，还有多篇论文探讨了模型驱动的深度学习架构设计方法，如[12]中提出了一种基于深度学习的网络的网络，称为 CsiNet，以减少大规模 MIMO 系统中的反馈开销。CsiNet 的网络架构是通过模仿 CS 架构获得的，CS 架构可以看

作是模型驱动 DL 的一个特例。CsiNet 主要包括一个卷积神经网络（CNN），该网络成功地进行了图像处理，并采用了一种自动编码器架构，该架构包括一个用于压缩感知的编码器和一个用于重建的解码器。每个细化网络单元遵循残余网络的思想，即它将较浅层的输出传输到较深层的输入，以避免 DNN 中的梯度消失问题。但该网络及其改进网络 CsiNet-LSTM[13] 不适用于实际的时变信道，因为线性全连接网络不适合描述时间相关性。另外该设计也未考虑天线的空间相关性。对于信道估计这一问题，数学模型难以精确建模描述信道的时间相关性与频域相关性，从而使得基于模型驱动的深度学习架构容易忽略数据的相关关系，进而限制模型的性能。

# 4 研究现存问题

目前已有的基于深度学习的信道估计算法已经有对于 MIMO 系统的信道有良好的性能表现，但是仍然存在部分问题。例如基于导频估计信道的多种深度学习算法，由于需要从少量时频信道值获得整个时频域的信道值，首先需要使用诸如插值等手段重建高精度信道矩阵，在重建过程中会将噪声扩散至其他时频位置，而非线性插值会使得扩散后的噪声非高斯白噪声，从而导致用于处理高斯白噪声的深度卷积网络表现不佳。另外，现有网络存在不够精细的缺点，虽然在信噪比较低的情况下能够取得明显优于传统算法的特点，但是当信噪比较高时，性能出现了一定的退化。这是由于传统深度学习，如 DNN 或 CNN，存在一定的精细度的限制，为了提高深度学习在这些情形下对信道的估计能力，需要完成进一步的研究，采用新的结构或者设计更加有效的网络架构。最后，部分设计忽视了实际通信信道中的时间相关性或者频率相关性，难以应用至现实信道。

# 5 研究展望

近年，深度学习在 CV 以及自然语言处理（NLP）领域取得了引人瞩目的发展。在自然语言处理领域提出了具有重要意义的 Transformer 模型[14]，该模型弥合了 CV 与 NLP 之间的模型差异，且具有应用至多个领域的潜力，目前已有相关尝试[15]。该模型可以观察到两个维度的信息，如时间的相关性和特征的相关

性，在信道估计中可以用于建模时频两个维度的相关性。在 5G 时代，信道的时间与频率选择性衰落效应更加显著，传统算法越来越难以对信道状态进行建模估计，在目前的通信系统架构下，深度学习拥有的非线性拟和能力将会赋能通信。而且硬件的发展使得在基站等终端部署深度学习算法的成本降低，从而开发通信领域的深度学习算法将会成为未来的一个研究重点。另外，深度学习也有助于我们设计更加智能的通信系统，提高通信效率，突破传输速率的限制。

# 6 参考文献

[1] D. Neumann, T. Wiese, and W. Utschick, “Learning the MMSE channel estimator,” IEEE Trans. on Signal Process., vol. 66, no. 11, pp. 2905–2917, 2018.
[2] Z. Du, X. Song, J. Cheng, and N. C. Beaulieu, “Maximum likelihood based channel estimation for macrocellular OFDM uplinks in dispersive time-varying channels,” IEEE Trans. on Wireless Commun., vol. 10, no. 1, pp. 176–187, 2011.
[3] C. Jiang, H. Zhang, Y. Ren, Z. Han, K.-C. Chen, and L. Hanzo, “Machine learning paradigms for next-generation wireless networks,” IEEE Wireless Commun., vol. 24, no. 2, pp. 98–105, 2017.
[4] Er, M.J., Zhou, Y.: Theory and novel applications of machine learning. InTech, 2009.
[5] H. He, C.-K. Wen, S. Jin, and G. Y. Li, “Deep learning-based channel estimation for beamspace mmWave massiveMIMOsystems,” IEEE Wireless Commun. Lett., vol. 7, no. 5, pp. 852–855, 2018.
[6] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, “Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising,” IEEE Trans. Image Process., vol. 26, no. 7, pp. 3142–3155, 2017.
[7] P. Dong, H. Zhang, G. Y. Li, I. S. Gaspar, and N. NaderiAlizadeh, “Deep CNN-based channel estimation for mmWave massive MIMO systems,” IEEE J. Sel. Topics in Signal Process., vol. 13, no. 5, pp. 989–1000, 2019.
[8] C.-J. Chun, J.-M. Kang, and I.-M. Kim, “Deep learning-based joint pilot design and channel estimation for multiuser MIMO channels,” IEEE Commun. Lett., vol. 23, no. 11, pp. 1999–2003, 2019.
[9] Y. Yang, F. Gao, X. Ma, and S. Zhang, “Deep learning-based channel estimation for doubly selective fading channels,” IEEE Access, vol. 7, pp. 36 579–36 589, 2019.
[10] M. Soltani, V. Pourahmadi, A. Mirzaei, and H. Sheikhzadeh, “Deep learning-based channel estimation,” IEEE Commun. Lett., vol. 23, no. 4, pp. 652–655, 2019.
[11] C. Dong, C. C. Loy, K. He, and X. Tang, “Image super-resolution using deep convolutional networks,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 38, no. 2, pp. 295–307, 2016.
[12] C. K. Wen, W. T. Shih, and S. Jin, “Deep Learning for Massive MIMO CSI Feedback,” IEEE Wireless Commun. Lett., vol. 7, no. 5, 2018, pp. 748–51.
[13] T. Wang, C. Wen, S. Jin, G. Y. Li, "Deep Learning-Based CSI Feedback Approach for Time-Varying Massive MIMO Channels," IEEE Wireless Commun. Lett., Vol. 8, no. 2, pp. 416-419, 2019.
[14] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In NeurIPS, 2017.
[15] Y. Xu, M. Yuan, and M.-O. Pun, “Transformer empowered CSI feedback for massive MIMO systems,” in Wireless and Optical Commun. Conf., Taipei, Taiwan, Oct. 2021, pp. 157–161.

# 二、开题报告

## 1 问题背景

### 1.1 通信系统与信道

第五代无线接入技术 (5G)，被称为 New Radio (NR)，将解决各种使用场景，从增强的移动宽带到超可靠的低延迟通信，再到大规模机器式通信。关键技术特点包括超精益传输，支持低延迟，先进的天线技术，频谱灵活性，包括在高频段操作和高低频频段之间的相互工作，是未来通信发展的风向标。而无线通信系统的性能很大程度上受到无线信道的影响，接收机与发射机之间的传播路径非常复杂，信道估计的精度将直接影响整个系统的性能，从而其是一项非常重要的任务。在 New Radio 标准下，通信系统涵盖多类物理信道，如物理随机接入信道 (PRACH)、物理下行共享信道 (PDSCH)、物理上行共享信道 (PUSCH)、物理下行控制信道 (PDCCH)、物理上行控制信道 (PUCCH)等。其中 PDSCH 是 LTE 中主要承载用户数据的下行链路通道，在系统中有重要意义。

而在实际的通信系统中，任何信号在传输过程中均会受到由于不理想信道带来的污染，多种噪声会加在信号本身，给接收端的解调以及检测等工作带来很大的阻碍。为了能够提高通信系统的效率，我们需要尽可能消除不理想信道导致的信号扭曲，而这就需要我们对信道的特征进行描述，也就是信道估计。通常信道估计是通过比较已知信号在收发机两端的不同来获得大致的信道矩阵。整体过程为：首先在发射机端，我们发送一系列已知参考信号，即导频信号，这些导频信号通常经过一定的设计使其便于在接收端检测得到；随后这些信号通过信道被噪声扭曲；最后我们在接收机端收到相关信号，通过比较收发信号的差异，我们得到其相关关系，从而得到对信道的估计。

### 1.2 参考信号

参考信号，即为导频信号，是发射端提供给接收端用于信道估计或者信道探测的一种已知信号。在收到信号之后，若已知信道信息，即可解调接收信号得到发射信号。对于物理下行共享信道，常用解调参考信号信号即解调参考信号

(Demodulation Reference Signal，DMRS) 进行估计以及相关解调。解调参考信号在时频平面上占有少量的位置，并随数据信号在发射机发送。接收机知道解调参考信号的时频位置和值。在接收端，通过比较发送和接收的解调参考信号，得到解调参考信号子信道矩阵。解调参考信号信道估计的目的是基于解调参考信号子信道矩阵得到各时频位置的信道响应，并恢复全信道矩阵。由于解调参考信号的密度较低、且由于 5G 的应用需要面对高动态的应用场景，单个符号内的信道变化可能会越来越严重，传统的线性恢复算法，如维纳滤波等，在实际部署时仍然有效但不是最优算法。为了提高信道估计的精度与准确性，以及提高信道估计过程中，系统对噪声的抑制能力，我们需要设计更加有效的算法。深度学习是一个复杂的机器学习算法，在语音和图像识别方面取得的效果，远远超过先前相关技术。与传统方法相比，基于深度学习的技术的优点之一是其鲁棒性。尤其是当数据量较大时，深度学习具有优于传统方法的描述数据特征的良好表征能力。

## 1.3 相关工作

最近几年内，深度学习在多个领域取得重大突破，在通信领域也得到了广泛关注。其优秀的非线性拟合能力以及学习能力使其能够从数据中学习到一定的规律，获得比传统方法更优异的性能，如准确性、鲁棒性。在通信领域已有多个课题运用深度学习，包括信道估计、信号检测、编码与解码等。在数据传输速率与带宽不断增大的发展趋势下，深度学习具有良好的应用前景。对于本研究题目，即基于深度学习的信道估计，已有论文进行讨论。如[1][2]和[3]针对大型 MIMO 多用户系统提出了基于经典深度学习的结构，即线性层或卷积层的组合，提出了包括 LDAMP[1]、SFT-CNN[2] 等深度学习结构，取得了大幅优于超过经典算法，如 LMMSE（线性最小均方）、EM（最大期望）[4][5]的效果。同时如 SPR-CNN[2]等结构有助于压缩导频长度，节约宝贵的频谱资源。随着深度学习在图像处理领域的飞速发展，部分研究尝试将信道时频矩阵通过图像处理的方式进行估计[6]。部分研究尝试将专家知识嵌入深度学习结构，提出了基于理论模型的深度学习结构[1][7][8]。随着深度学习的发展，部分研究也在更新用于信道估计的深度学习结构[9]。

## 1.4 本研究的目的与意义

从深度学习受到热切关注开始，已有大量研究利用深度学习进行信道估计算法设计，但是限制于深度学习的结构特点以及通信领域的数据特点，这些研究存在部分问题，如难以同时关注信道的频域相关性和时域相关性；由于参考信号的稀疏性，在恢复高精度时频信息时，会将参考信号时频位置的噪声扩散至数据位置；而基于模型的深度学习结构会受模型精度的影响，从而限制其性能 。本研究旨在利用新进发展的深度学习方法，设计信道估计深度学习架构，依靠深度学习优秀的非线性拟合能力，提高在噪声环境下通过解调参考信号估计信道信息的准确率以及可靠性。

该研究有利于大幅降低接收信号解调误码率、抵抗多径效应以及扩展 5G 信号的应用场景 (如高动态环境)等多方面，是一项具有现实意义以及启发性的任务，在研究领域以及工业界均有重要意义。

# 2 研究主要内容与技术路线

## 2.1 主要研究内容

本研究基于 2021 年无线通信 AI 大赛（Wireless Communication AI Competition）赛道 2：基于 AI 的信道估计。其基本内容如下：

针对 32 发送 2 接收天线（32T2R）的多天线信道环境下的 DMRS 信号设计相关的信道估计算法，使其具有良好的预测精度以及泛化能力。数据来自单小区单用户 32T2R 的下行多天线信道，数据样本数量为 21 万例样本。对于 12 个子载波和 14 个 OFDM 符号，DMRS 图样请见图 1.1 示。每个样本的输入为 H_in，维度为$48 \times 2 \times 2$，分别对应 48 个子载波，2 个符号，实部和虚部。每个样本的输出为 H_out，维度为$96 \times 14 \times 2$，分别对应 96 个子载波，14 个符号，实部和虚部。频域一共有 96 个子载波，H_out 占据所有子载波，而 H_in 只占据其中的奇数子载波。时域一共有 14 个 OFDM 符号，H_out 占据所有 OFDM 符号，而 H_in 只占据其中第 4 个和第 12 个 OFDM 符号。发送端采用波束赋形，将 32 根发送天线的信号转化为一路信号，所有子载波采用相同的波束赋形向量。接收端有两根接收天线，但 H_in 和 H_out 只代表第一根接收天线的信号。每个样本

的频域存在相关性，时域也同样存在相关性。需要充分利用该相关性以获得良好的预测性能。对于每个信道样本，信道相关参数均为随机生成，如时延扩展、用户移动速度、信噪比等，其中仅有信噪比已知。

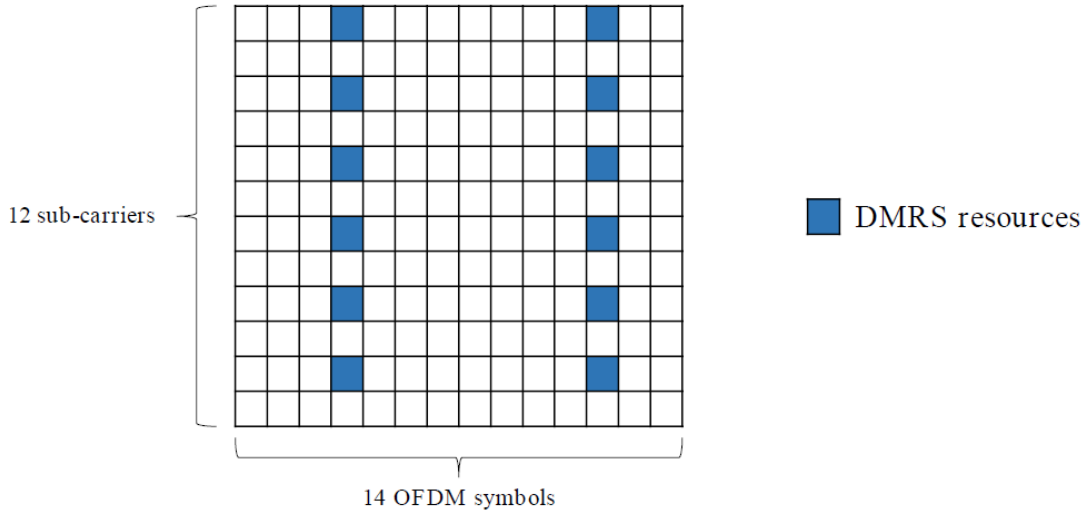



图 1.1 DMRS 时频矩阵样例

该研究的主要内容即为利用接收机接收到的 DMRS 信号 H_in 与实际收到的信号 H_out 设计相应的深度学习算法来模拟信道估计过程，从而尽可能准确地由收到的 DMRS 信号恢复得到实际收到的信号。其准确性由恢复信号与实际信号的归一化均方误差来量度。同时在后续研究中，我们可以进一步考虑如计算复杂度等因素，设计满足准确性要求的算法同时降低计算量，从而可以将算法部署至移动终端，达到提高用户的通信质量等目的。

## 2.2 技术路线

### 2.2.1 基础模型

深度学习近年来在自然语言处理（NLP）以及计算机视觉（CV）取得了引人注目的成就。在 2017 年提出的 Transfomer 模型[10]，其结构如图 2.1 所示。该基础模型由两个部分组成，图中左半部分的编码器（Encoder）与图中右半部分的解码器（Decoder），可以将其简单理解为具有注意力机制的 Autoencoder。其基本工作机制如下：在 Encoder 中，通过 Multi-Head Attention（多头注意力）模块建立时间维度的相关关系，将不同时间（第一个维度）的特征（第二个维度）联系起来，并且该模块强迫该模型对不同时刻的特征赋予不同比例的“注意力”。考虑

到信道的特点，该模块有助于时频图像的恢复。Feed Forward 模块则进一步提取特征信息。在 Decoder 中，大部分模块与 Encoder 相同或类似，而 Masked Multi-Head Attention 模块通过遮盖当前时刻之后的信息，使得训练的行为方式与预测的行为方式一致。而 Positional Encoding 用于加载前后关系的信息。Transformer 在 NLP 问题上取得了显著的效果，并且其优秀的可移植性使得 CV 等领域也有大量相关工作利用 Transformer 提取图片中的语义信息，并进行如分类、语义分割、图像重建等任务。对于本研究的具体问题而言，我们可以将时频矩阵视为一张二维图像，则同样可以利用 CV 的思路进行处理。

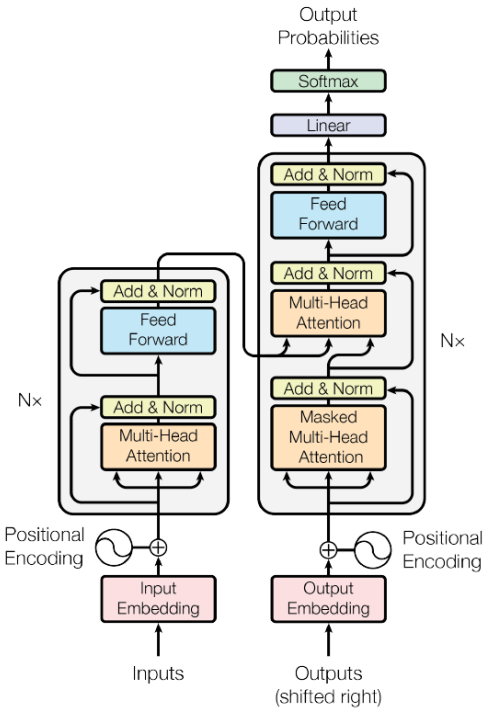



图 2.1 Transformer 结构

### 2.2.2 去噪与重建

Transformer 在 CV 已经得到大量运用。而针对信道估计问题，其中重点为对信道矩阵进行去噪以及根据稀疏的已知信道重建整个信道信息。根据之前叙述，我们可将该问题转化为 CV 问题，则我们需要讨论如何利用 Transformer 进行去噪以及图像重建。在此仅简单叙述可运用至该问题的具体模型。

对于去噪模型，2021 年中科大团队利用 Transformer 建立了 Uformer 模型[11]，如图 2.2，其仿造 U-Net 架构，将特征空间维度扩大再压缩，同时特征空间尺度压缩再放大，从而提取出主要信息，抑制噪声信息。该方法在多个具体问题上取

得了最佳指标，如去除摩尔纹、去雨等，且具有更低的计算复杂度。

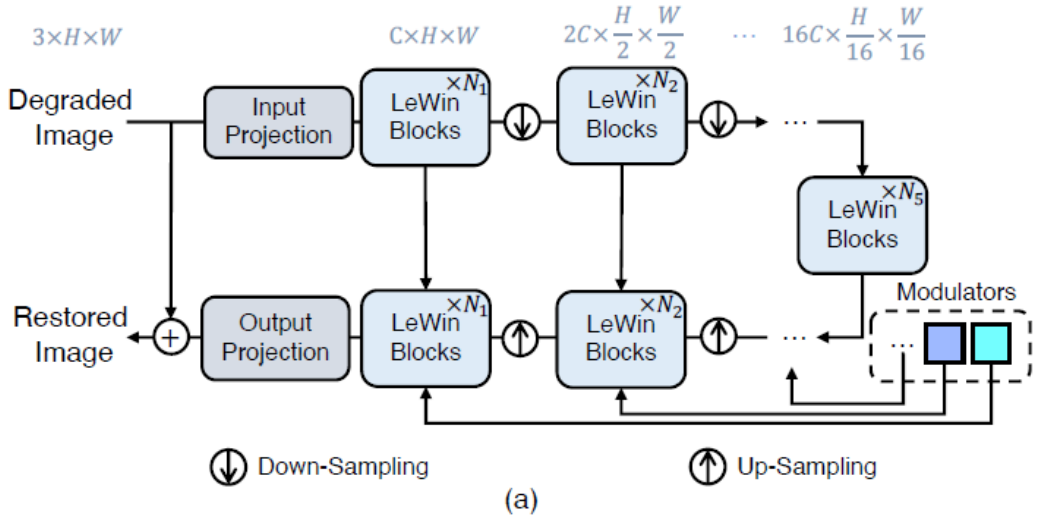



图 2.2 Uformer 基本结构

对于重建模型 Transformer 同样也取得了引人注目的成绩。最近 FAIR 团队一篇论文提出了 Masked Autoencoders（MAE）[12]，能够从被大范围遮盖的残留图像恢复原始图像语义。如图 2.3 中结构，MAE 以未遮掩图像作为输入，经过 Encoder 编码提取语义信息，随后输入 Decoder 进行图像重建，其中 Encoder 和 Decoder 均由 Transformer 构成，最终能有效恢复图像。该模型在多个数据集上取得了 SOTA 结果。

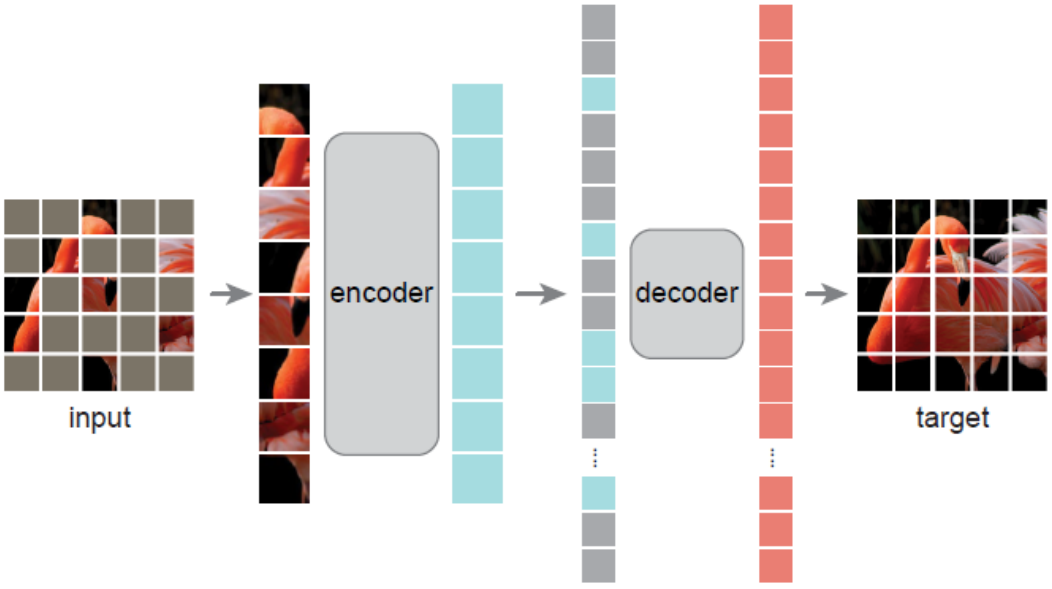



图 2.3 MAE 总体结构

Transformer 能够有效完成去噪与重建的工作，且有大量工作作为

借鉴，为迁移学习提供帮助。

### 2.2.3 问题迁移

在此我们将讨论利用 Transformer 进行信道估计的可能性以及一些注意点。对于信道时频矩阵，其每一个时刻与其他时刻的相关性不同，不同频段的频域相关性也不同，而 Transformer 中的注意力机制有利于深度学习模型寻找到该时频相关性，而信道估计的重点就在于利用时频相关性。同时，利用上述基于 Transformer 的去噪器以及重建器，信道噪声可以被抑制，同时可以获得相比传统深度学习更加精确的信道估计结果。需要注意的是信道矩阵的尺度与传统图像处理不同，我们需要针对具体情况进行设计，以达到最佳效果。例如，可将信道的频谱信息作为初始特征，则该问题又可以视为 NLP 问题；另外，去噪器与恢复器的前后顺序也需要注意，例如先进行信道恢复可能会带来噪声扩散的不良影响；最后需要考虑如计算复杂度的问题，在完成信道估计的同时节省计算开支，以便于部署。

由此，本研究的基本设计思路为：利用 Transformer 构造去噪器或恢复器（或两者），对 DMRS 信道矩阵进行处理，并重建整个信道矩阵。可参考 U-Net 或 Autoencoder 架构。我们将重点关注如何利用注意力机制来进行架构的设计，从而使得 DMRS 信号的时频特征均有得到良好利用。

如图 2.4，搭建简易 Autoencoder 进行仿真实验。从图中可以看到，当网络结构非常简单，仅使用数层线性层堆叠时，仍然可以在相当程度上利用 DMRS 信号估计得到整个信道矩阵。其缺点在于不够精确，且有明显的噪声污染情况。结合对于 Transformer 与 Autoencoder 的相似性以及对其强大拟合能力的讨论，利用 Transformer 搭建信道估计网络是切实可行的。

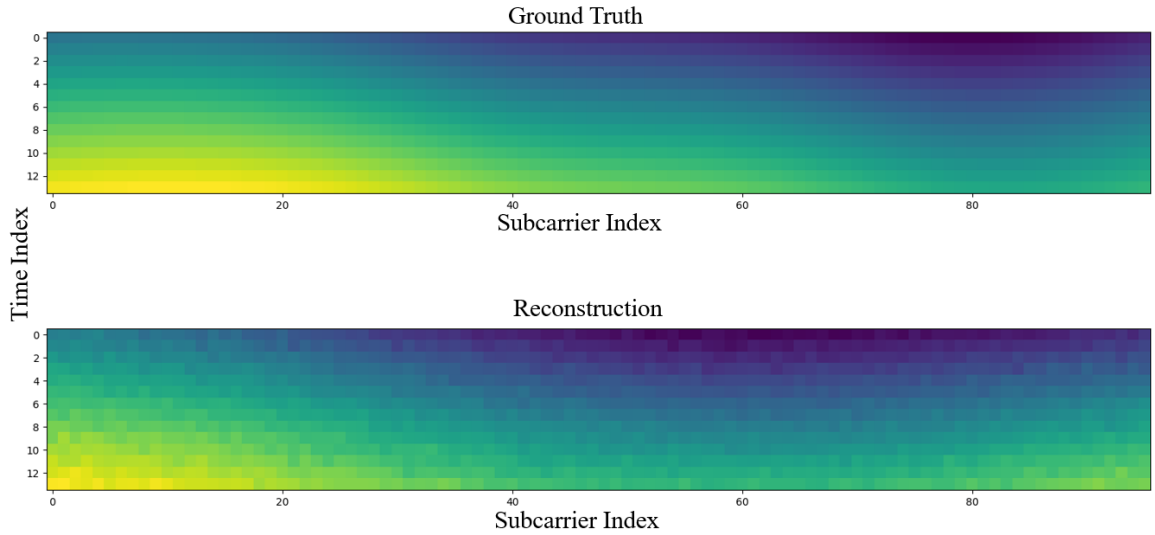



图 2.4 简易 Autoencoder 仿真结果

# 3 进度安排

2021-11-08 至 2021-11-21：查阅相关资料，调研国内外研究现状。

至 2021-11-28：完成外文翻译

2021-11-29 至 2021-12-19：熟悉 DMRS 信号及其信道特点。

2021-12-20 至 2022-01-02：撰写文献综述、文献翻译和开题报告，准备开题答辩。

2022-01-03 至 2022-01-07：开题答辩。

2022-01-15 至 2022-02-17：寒假。

2022-02-21 至 2022-03-06：完成 DMRS 信道估计方法研究并进行仿真分析。

2022-03-07 至 2022-03-20：完成已有基于 AI 估计 DMRS 信道的研究。

2022-03-21 至 2022-04-17：完成新的或改进基于 AI 估计 DMRS 信道方案的设计，与已有 AI 设计方案进行比较。

2022-04-18 至 2022-05-01：构建实验，完成数据采集与分析，对结果进行分析比较验证

2022-05-02 至 2022-05-15：撰写、修改并上传毕业论文。

2022-05-16 至 2022-05-24：查重、论文答辩。

# 4 预期目标

1、复现基于经典深度学习的信道估计算法。

2、提出基于 Transformer 的信道估计深度学习结构与方法。

3、基于实际的通信信道数据，仿真验证提出方法的有效性，实现优于经典深度学习的效果。

4、拓展研究其他先进的基于深度学习的信道估计方法并仿真验证其可行性。

# 5 参考文献

[1] H. He, C.-K. Wen, S. Jin, and G. Y. Li, “Deep learning-based channel estimation for beamspace mmWave massive MIMO systems,” IEEE Wireless Commun. Lett., vol. 7, no. 5, pp. 852–855, 2018.
[2] P. Dong, H. Zhang, G. Y. Li, I. S. Gaspar, and N. NaderiAlizadeh, “Deep CNN-based channel estimation for mmWave massive MIMO systems,” IEEE J. Sel. Topics in Signal Process., vol. 13, no. 5, pp. 989–1000, 2019.
[3] C.-J. Chun, J.-M. Kang, and I.-M. Kim, “Deep learning-based joint pilot design and channel estimation for multiuser MIMO channels,” IEEE Commun. Lett., vol. 23, no. 11, pp. 1999–2003, 2019.
[4] Z. Du, X. Song, J. Cheng, and N. C. Beaulieu, “Maximum likelihood based channel estimation for macrocellular OFDM uplinks in dispersive time-varying channels,” IEEE Trans. on Wireless Commun., vol. 10, no. 1, pp. 176–187, 2011.
[5] C. Jiang, H. Zhang, Y. Ren, Z. Han, K.-C. Chen, and L. Hanzo, “Machine learning paradigms for next-generation wireless networks,” IEEE Wireless Commun., vol. 24, no. 2, pp. 98–105, 2017.
[6] M. Soltani, V. Pourahmadi, A. Mirzaei, and H. Sheikhzadeh, “Deep learning-based channel estimation,” IEEE Commun. Lett., vol. 23, no. 4, pp. 652–655, 2019.
[7] C. K. Wen, W. T. Shih, and S. Jin, “Deep Learning for Massive MIMO CSI Feedback,” IEEE Wireless Commun. Lett., vol. 7, no. 5, 2018, pp. 748–51.
[8] T. Wang, C. Wen, S. Jin, G. Y. Li, "Deep Learning-Based CSI Feedback Approach for Time-Varying Massive MIMO Channels," IEEE Wireless Commun. Lett., Vol. 8, no. 2, pp. 416-419, 2019.
[9] Y. Xu, M. Yuan, and M.-O. Pun, “Transformer empowered CSI feedback for massive MIMO systems,” in Wireless and Optical Commun. Conf., Taipei, Taiwan, Oct. 2021, pp. 157–161.
[10] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In NeurIPS, 2017.
[11] Zhendong Wang, Xiaodong Cun, Jianmin Bao, and Jianzhuang Liu. Uformer: A general U-shaped Transformer for image restoration. arXiv:2106.03106, 2021.
[12] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollar, and Ross Girshick. Masked autoencoders are scalable vision learners. arXiv preprint arXiv:2111.06377, 2021.

# 三、外文翻译

## 0.摘要

对于毫米波（mmWave）大规模多输入多输出（MIMO）系统，通常使用混合处理架构来降低复杂性和成本，这在信道估计中提出了非常具有挑战性的问题。本文采用深度卷积神经网络（CNN）来解决这个问题。我们首先提出了一种利用空间和频率相关性的基于空间频率 CNN（SF-CNN）的信道估计，其中相邻子载波处的损坏信道矩阵被同时输入到 CNN 中。然后，利用时变信道中的时间相关性，开发了基于空间频率-时间 CNN（SFT-CNN）的方法以进一步提高准确性。此外，我们设计了一种空间导频减少 CNN（SPR-CNN）以节省信道估计的空间导频开销，其中几个连续相干间隔中的信道被具有存储器的信道估计单元分组和估计。数值结果表明，所提出的基于 SF-CNN 和 SFT-CNN 的方法优于非理想最小均方误差（MMSE）估计器，且复杂度降低，而且性能接近难以实际部署的理想 MMSE 估计器。它们对不同的传播场景也很有效。基于 SPR-CNN 的方法达到了与基于 SF-CNN 和 SFT-CNN 的方法相当的性能，同时仅以复杂为代价仅需要约三分之一的导频开销。本文的结果清楚地表明，深卷积网络可以有效地利用信道相关性，以提高 mmWave 大规模 MIMO 系统的估计性能。

## 1.引文

略

传统的信道估计方法在实际情况较复杂的信道模型中往往表现不佳，而且复杂度高。相比之下，深度卷积神经网络(deep convolutional NN,CNN)更能从大量的数据中提取出隐藏在信道矩阵下的固有特征，并通过高效的并行计算方法提供了以更低的复杂度更准确地估计信道的潜力。在本文中，我们使用深度 CNN 来解决 mmWave 大规模 MIMO-OFDM 系统的信道估计问题。为了利用 OFDM 中相邻子载波上信道之间的相关性，我们首先提出了一种基于空间频率 CNN(SF-CNN) 的信道估计方法，其中在相邻子载波上初步估计的信道矩阵同时输入到 CNN[1] 中。为了进一步利用时间相关性，提出了一种基于空间-频率-时间 CNN(SFT-CNN)的信道估计方法，该方法利用先前相干区间的信道信息来估计当

前相干区间的信道矩阵。基于 SFT-CNN 的方法以一种简单的方式整合了所有类型的信道相关，并产生了显著的性能增益，可用于显著节省大规模阵列的空间导频开销。因此，我们提出了一种基于空间导频缩减 CNN(SPR-CNN)的信道估计方法，该方法将几个连续相干间隔内的信道进行分组，并由一个带记忆的信道估计单元(CEU)进行估计。从数值结果来看，所提出的 SF-CNN 和基于 SFT-CNN 的方法优于非理想最小均方误差(MMSE)估计，性能非常接近实际系统中难以实现的理想 MMSE 估计。它们也比 MMSE 估计器具有更低的复杂度，并且在面对不同的信道统计量时表现出独特的鲁棒性来保持相当好的性能。基于 SPR-CNN 的方法通过仅使用大约三分之一的空间试验开销和适度增加的复杂性，实现了与 SF-CNN 和基于 SFT-CNN 方法相当的性能。

本文的其余部分组织如下。第二部分描述了考虑的 mmWave 大规模 MIMO 系统，然后在第三部分中提出了基于 SF-CNN 的信道估计。第四部分进一步发展了基于 SFT-CNN 和基于 SPR-CNN 的信道估计。第五节给出了数值结果，最后第六节为结论。符号:在本文中，我们分别用大写黑体字和小写黑体字来表示矩阵和向量。$||\cdot||_F$，$(\cdot)^T$，$(\cdot)^H$，$(\cdot)^{-1}$，和$\mathbb{E}\{\cdot\}$分别表示 Frobenius 范数、转置、共轭转置、逆和期望。$\mathcal{CN}(\mu,\sigma^2)$表示圆形对称复高斯分布，其中$\mu$为平均值、$\sigma^2$为方差。$\delta(\cdot)$和$\lceil\cdot\rceil$表示 delta 函数和 ceiling 函数。$\mathbb{N}_+$表示所有正整数的集合。

# 2.系统模型

我们考虑一个毫米波大规模 MIMO-OFDM 系统，如图 1 所示，其中发射机有$N_T$条天线和$N_T^{RF}$条射频链路，而接收机有$N_R$条天线和$N_R^{RF}$条射频链路。移相器用于在发射器和接收器侧将大量天线与更少数量的 RF 链连接。我们因此假设$N_T \gg N_T^{RF}$以及$N_R \gg N_R^{RF}$。

根据[10]，延迟域中接收机和发射机之间的$N_R \times N_T$信道矩阵由下式给出：

$$\mathbf{H}(\tau) = \sqrt{\frac{N_T N_R}{L}} \sum_{l=1}^{L} \alpha_l \, \delta(\tau - \tau_l) \mathbf{a}_R(\varphi_l) \mathbf{a}_T^H(\phi_l), \tag{1}$$

其中$L$是主要路径的数量，$\alpha_l \sim \mathcal{CN}(0,\sigma_\alpha^2)$是第 l 条路径的传播增益，$\sigma_\alpha^2$是其平均功率增益。$\tau_l$是第 l 条路径的延迟，$\varphi_l$和$\phi_l \in [0,2\pi]$分别是发射机与接收机到达与离开时的方位角（AoA/AoD）。对于均匀线阵（ULA），对应的响应向量可以表示为：

$$a_R(\varphi_l) = \frac{1}{\sqrt{N_R}}\left[1, e^{-j2\pi\frac{d}{\lambda}sin(\varphi_l)}, \dots, e^{-j2\pi\frac{d}{\lambda}(N_R-1)\,sin(\varphi_l)}\right]^T, \quad (2)$$

$$a_T(\phi_l) = \frac{1}{\sqrt{N_T}}\left[1, e^{-j2\pi\frac{d}{\lambda}sin(\phi_l)}, \dots, e^{-j2\pi\frac{d}{\lambda}(N_T-1)\,sin(\phi_l)}\right]^T, \quad (3)$$

其中$d$和$\lambda$分别表示相邻天线之间的距离与载波波长。

根据(1)中的信道模型，给出 OFDM 中第 k 个子载波的频域信道为

$$\mathbf{H}_k = \sqrt{\frac{N_T N_R}{L}}\sum_{l=1}^{L}\alpha_l\, e^{-j2\pi\tau_l f_s\frac{k}{K}}\mathbf{a}_R(\varphi_l)\mathbf{a}_T^H(\phi_l), \quad (4)$$

其中 fs 为采样率，K 为 OFDM 子载波数。

为了估计$\mathbf{H}_k$，发射机仅激活一个射频链以在每个信道使用的一个波束上发射导频，而接收机通过使用与不同波束相关联的所有射频链来组合接收的导频。更详细地说，发射机使用$M_T$个波束形成向量，$\mathbf{f}_{k,u} \in \mathbb{C}^{N_T\times 1}, u = 1, \dots, M_T$传输导频$x_{k,u}$。对于每个波束形成向量$\mathbf{f}_{k,u}$对应的发射导频信号，接收机采用$M_R$个组合向量，$w_{k,v} \in \mathbb{C}^{N_R\times 1}, v = 1, \dots, M_R$，分别对其进行处理。由于接收器配备了$N_R^{RF}(< M_R)$条射频链，它只能在一个信道使用中使用$N_R^{RF}$个组合矢量。然后，如果接收端使用所有$M_R$个向量来处理一个携带导频的波束形成向量，则需要使用的信道数量为$\left\lceil\frac{M_R}{N_R^{RF}}\right\rceil$.1。则与接收机基带第 k 个子载波相关联的导频信号矩阵可写成

$$\mathbf{Y}_k = \mathbf{W}_k^H\mathbf{H}_k\mathbf{F}_k\mathbf{X}_k + \widetilde{\mathbf{N}}_k, \quad (5)$$

其中$\mathbf{W}_k = \left[\mathbf{w}_{k,1}, \dots, \mathbf{w}_{k,M_R}\right]$和$\mathbf{F}_k = \left[\mathbf{f}_{k,1}, \dots, \mathbf{f}_{k,M_T}\right]$分别是结合矩阵和波束形成矩阵，$\mathrm{X}_k$是$M_T \times M_T$的对角阵其第 u 个元素为$x_{k,u}$。$\widetilde{\mathbf{N}}_k = \mathbf{W}_k^H\mathrm{N}_k$表示合并后的有效噪声，$\mathrm{N}_k$为合并前元素参数为$\mathcal{CN}(\mu, \sigma^2)$的加性高斯白噪声(AWGN)。

我们在频域和时域都考虑导频插入。具体而言，相邻$Q(Q \geq 2)$个子载波分别在相干间隔的开始位置放置长度相同的导频，形成导频子载波块，每个相干间隔中的其余时隙用于数据传输。两个导频子载波块由专用于数据传输的$Q_d(Q_d \geq 0)$个子载波分隔。导频用于估计相应时频位置的信道。在已估计信道的基础上，利用插值方法在没有导频的情况下得到各位置的信道。很明显，插值的准确性取决

[1] 这个导频传输过程可以捕获毫米波通道中的主要路径。虽然在发射机同时激活多个具有不同波束的射频链可以加速导频传输过程，但无法捕获主路径，导致信道估计性能较差。因此，在大量的相关工作中，在每个信道使用过程中，发射机的一个波束上只激活一个射频链来传输导频，以保证信道估计算法[9]、[11]的性能。同样，在本文中，我们考虑的导频传输模式也有助于基于 CNN 的信道估计达到很好的精度。

于估计的通道的准确性和通道的变化。因此，我们在本文中将重点研究如何提高基于导频的信道估计的精度，为插值提供更可靠的参考值。

# 3.基于 SF-CNN 的信道估计

在本节中，我们首先阐述基于 SF-CNN 的信道估计，包括所提出的方法的概述，SF-CNN 的离线训练，以及在线部署。然后分析了在线估计的计算复杂度。

## 3.1.算法描述

1）信道估计过程: 图 1 说明了邻接$Q(=2)$个子载波$k_0$和$k_0+1$的信道估计过程，以简化我们的讨论，即使扩展到有$Q(>2)$的情况是很简单的。在不损失一般性的前提下，假设$\mathbf{W}_k=\mathbf{W}$, $\mathbf{F}_k=\mathbf{F}$，且$\mathbf{X}_k=\sqrt{P}\mathbf{I}$对于$k\in\{k_0,k_0+1\}$，其中$P$表示发射功率。导频信号矩阵$\mathbf{Y}_k$变成

$$\mathbf{Y}_k=\sqrt{P}\mathbf{W}^H\mathbf{H}_k\mathbf{F}+\widetilde{\mathbf{N}}_k \tag{6}$$

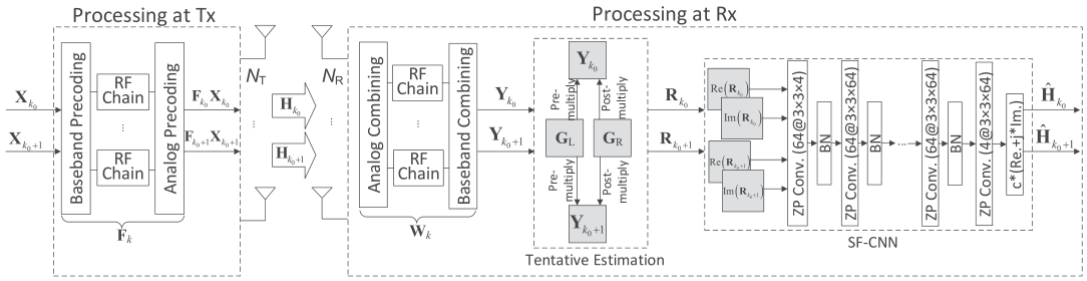



Fig. 1. SF-CNN based channel estimation.

然后，$\mathbf{Y}_k$通过暂测估计(TE)模块，该模块使用两个矩阵$\boldsymbol{G}_L$与$\boldsymbol{G}_R$处理$\mathbf{Y}_k$，并输出$\mathbf{H}_k$的粗略估计，即:

$$\mathbf{R}_k=\mathbf{G}_L\mathbf{Y}_k\mathbf{G}_R=\sqrt{P}\mathbf{G}_L\mathbf{W}^H\mathbf{H}_k\mathbf{F}\mathbf{G}_R+\mathbf{G}_L\widetilde{\mathbf{N}}_k\mathbf{G}_R, \tag{7}$$

其中

$$\mathbf{G}_L=\begin{cases}\mathbf{W}^H, & M_R<N_R\\ (\mathbf{W}\mathbf{W}^H)^{-1}\mathbf{W}, & M_R\geq N_R\end{cases} \tag{8}$$

且

$$\mathbf{G}_R=\begin{cases}\mathbf{F}^H, & M_T<N_T\\ \mathbf{F}^H(\mathbf{F}\mathbf{F}^H)^{-1}, & M_T\geq N_T\end{cases}$$

然后将初步估计的信道矩阵$\mathbf{R}_{k_0}$和$\mathbf{R}_{k_0+1}$同时输入到 SF-CNN 中，SF-CNN 通过映

射关系输出估计的信道矩阵$\widehat{\mathbf{H}}_{k_0}$和$\widehat{\mathbf{H}}_{k_0+1}$

$$\left\{\widehat{\mathbf{H}}_{k_0},\widehat{\mathbf{H}}_{k_0+1}\right\}=f_{\Phi}\left(\mathbf{R}_{k_0},\mathbf{R}_{k_0+1};\Phi\right), \tag{10}$$

其中$\Phi$代表 SF-CNN 的参数集合。

2）SF-CNN 离线训练: 对于提出的 SF-CNN，在仿真环境中根据一定的信道模型生成由$N_{tr}$个样例组成的训练集，其中$(\underline{\mathbf{R}}_i,\underline{\mathbf{H}}_i)$表示第 i 个样本，其中$\underline{\mathbf{R}}_i$为输入数据，$\underline{\mathbf{H}}_i$为目标数据。$\underline{\mathbf{R}}_i\in\mathbb{C}^{N_R\times N_T\times 2}$是由$\mathbf{R}^i_{k_0'},\mathbf{R}^i_{k_0'+1}\in\mathbb{C}^{N_R\times N_T}$组成的三维矩阵，分别为子载波$k'_0$和$k'_{0+1}, k'_0\in\{1,2,\cdots,K-1\}$。$\underline{\mathbf{H}}_i\in\mathbb{C}^{N_R\times N_T\times 2}$也是一个三维矩阵，含$\frac{\mathbf{H}^i_{k_0'}}{c},\frac{\mathbf{H}^i_{k_0'+1}}{c}\in\mathbb{C}^{N_R\times N_T}$，其中$\mathbf{H}^i_{k_0'}$和$\mathbf{H}^i_{k_0'+1}$是对应的真实信道矩阵。$c>0$是一个缩放常数，使得所有目标数据$\underline{\mathbf{H}}_i$的实部和虚部的取值范围匹配 SF-CNN 输出层应用的激活函数。然后$\underline{\mathbf{R}}_i$输入 SF-CNN 来近似对应的缩放真实信道值$\underline{\mathbf{H}}_i$。

对于 mmWave 大规模 MIMO 系统，以$N_{\mathrm{T}}=32,\ N_{\mathrm{R}}=16$为例。如图 1 所示，SF-CNN 接收到初步估计的复信道矩阵$\mathbf{R}^i_{k_0'}\in\mathbb{C}^{16\times 32}$和$\mathbf{R}^i_{k_0'+1}\in\mathbb{C}^{16\times 32}$，作为输入，并将其实部和虚部分开，从而得到 4 个 16×32 实值矩阵。在随后的卷积层中，这四个矩阵通过 64 个 3×3×4 卷积滤波器和修正线性单元(ReLU)激活函数进行处理，生成 64 个 16×32 的实值矩阵。在处理每个特征矩阵时使用零填充(Zero padding, ZP)，使其卷积后维数保持不变。然后加入批归一化(batch normalization, BN)层，以避免梯度扩散和过拟合。对于接下来的八个卷积层，每个层使用 64 个 3×3×64 滤波器对前一层传递的特征矩阵进行 ZP 卷积，并输出 64 个 16×32 实值特征矩阵。对这 8 个层应用 ReLU 激活函数，每个层后面都有一个 BN 层。输出层使用 4 个 3×3×64 卷积录波器来处理 64 个 16×32 实值特征矩阵，并得到在$k'_0$和$k'_0+1$处子载波的缩放信道矩阵估计值，即$\frac{Re\left(\widehat{\mathbf{H}}^i_{k_0'}\right)}{c},\frac{Im\left(\widehat{\mathbf{H}}^i_{k_0'}\right)}{c},\frac{Re\left(\widehat{\mathbf{H}}^i_{k_0'+1}\right)}{c}$和$\frac{Im\left(\widehat{\mathbf{H}}^i_{k_0'+1}\right)}{c}$。输出层使用双曲正切激活函数将输出映射到区间$[-1,1]$。在放大并结合相应的实部和虚部后，获得了维度为 16×32 的复数值估计信道矩阵$\widehat{\mathbf{H}}^i_{k_0'}$和$\widehat{\mathbf{H}}^i_{k_0'+1}$。表 1 列出了 SF-CNN 的详细架构。

TABLE I
ARCHITECTURE OF THE SF-CNN

| Layer | Type | Number of filters | Size of filters | Activation function |
|---|---|---|---|---|
| 1 | Input | - | - | - |
| 2∼10 | Conv. | 64 | $3 \times 3$ | ReLU |
| 11 | Output | $2Q$ | $3 \times 3$ | tanh |

对 SF-CNN 进行离线训练的目标是最小化 MSE 损失函数

$$\mathrm{MSE}_{\mathrm{Loss}} = \frac{1}{N_{\mathrm{tr}}c^2}\sum_{i=1}^{N_{\mathrm{tr}}}\sum_{q=1}^{2}\left\|\mathbf{H}_{k_0'+q-1}^{i} - \widehat{\mathbf{H}}_{k_0'+q-1}^{i}\right\|_F^2. \tag{11}$$

SF-CNN 体系结构的设计借鉴了基于 CNN 的图像处理，并考虑了具体的信道估计任务。使用 SF-CNN 进行信道去噪，因此我们设置每一层的特征图的大小为$N_{\mathrm{R}} \times N_{\mathrm{T}}$。九个卷积隐藏层被用来充分揭示信道的内在结构。根据[23]，我们采用多个非常小的卷积滤波器即 3×3，以低复杂度实现良好的信道估计性能。根据我们的模拟试验，进一步增加卷积层的数量或卷积滤波器的数量不会带来主要的性能改善，但会导致 CNN 训练和测试的复杂性更高。

3）线上网络部署: 离线训练后，在接收端部署 SF-CNN 和 TE 模块，通过联合处理导频矩阵 $\mathbf{Y}_{k_0}, \mathbf{Y}_{k_0+1}, \ldots, \mathbf{Y}_{k_0+Q-1}$，输出估计的信道矩阵 $\widehat{\mathbf{H}}_{k_0}, \widehat{\mathbf{H}}_{k_0+1}, \ldots, \widehat{\mathbf{H}}_{k_0+Q-1}$。如果实际的信道模型不同于用于生成训练集的模型，一个简单的解决方案是微调。但由于难以采集真实信道，因此采用估计的信道来代替。很明显，使用更大的功率或更长的导频序列会使估计的信道更接近真实的信道，然而，这增加了在线微调的开销。幸运的是，如 Section V 中的图 6 和图 7 所示，离线训练的 SF-CNN 对于大多数之前没有观察到的新信道统计数据是相当稳健的。这意味着进一步的在线微调只能提供微小的性能改进，因此可能没有必要。

## 3.2.复杂度分析

在这一小节中，我们分析了在测试阶段提出的基于 SF-CNN 的信道估计的计算复杂度，并将其与使用估计协方差矩阵的非理想 MMSE 进行比较。所需的浮点操作数(FLOPs)被用作度量标准。

TABLE II
SF-CNN PARAMETER SETTINGS

| $l$ | $M_{1,l}$ | $M_{2,l}$ | $F_l$ | $N_{l-1}$ | $N_l$ |
|---|---|---|---|---|---|
| 1 | 16 | 32 | 3 | 4 | 64 |
| $2 \sim 9$ | 16 | 32 | 3 | 64 | 64 |
| 10 | 16 | 32 | 3 | 64 | 4 |

对于所提出的方法，FLOPs 来自于(7)和 SF-CNN 中的 TE 模块处理。通过假设$M_{\mathrm{T}} = N_{\mathrm{T}}$和$M_{\mathrm{R}} = N_{\mathrm{R}}$，(7)中的矩阵乘积需要$C_{\mathrm{TE}} \sim \mathcal{O}\big(QN_{\mathrm{T}}N_{\mathrm{R}}(N_{\mathrm{T}} + N_{\mathrm{R}})\big)$ [24]次 FLOPs。根据[25]，SF-CNN 处理所需的 FLOPs 为$C_{\mathrm{SF-CNN}} \sim \mathcal{O}\big(\sum_{l=1}^{L_{\mathrm{c}}} M_{1,l}M_{2,l}F_l^2N_{l-1}N_l\big)$，其中$L_{\mathrm{c}}$为卷积层数$M_{1,l}$与$M_{2,l}$表示第 l 层输出的每个特征图的行数和列数，$F$ 表示第 l 层使用的滤波器的边长，$N_{l-1}$和$N_l$表示第 l 层输入和输出特征图的个数。具体来说，这些参数根据上述 SF-CNN 离线训练列在表 II 中。然后基于 SF-CNN 信道估计的计算复杂度如下

$$C_{\text{SF-CNN-CE}} \sim \mathcal{O}\big(QN_TN_R(N_T + N_R) + N_TN_R\sum_{l=1}^{L_c} F_l^2\, N_{l-1}N_l\big), \tag{12}$$

对于 MMSE 信道估计，首先需要进行最小二乘(LS)信道估计，从而导致$C_{\mathrm{LS}} \sim \mathcal{O}(QN_{\mathrm{T}}^2N_{\mathrm{R}}^2)$的 FLOPs。然后，基于每通道实现一次的 LS 信道估计计算信道协方差矩阵，如果考虑空间和频率信道统计，则需要$C_{\mathrm{MMSE},1} \sim \mathcal{O}(Q^2N_{\mathrm{T}}^2N_{\mathrm{R}}^2)$的计算复杂度。最后，利用协方差矩阵对 LS 信道估计进行细化，相应的 FLOPs 为$C_{\mathrm{MMSE},2} \sim \mathcal{O}(Q^3N_{\mathrm{T}}^3N_{\mathrm{R}}^3)$。因此，MMSE 的总体计算复杂度为

$$C_{\mathrm{MMSE}} \sim \mathcal{O}(Q^3N_{\mathrm{T}}^3N_{\mathrm{R}}^3) \tag{13}$$

一般来说，很难直接比较$C_{\text{SF-CNN-CE}}$与$C_{\mathrm{MMSE}}$，因为前者除了$Q$，$N_T$和$N_R$以外依赖于$L_c$、$F_l$、$N_{l-1}$和$N_l$。如果$N_T = 32, N_R = 16, Q = 2$等 SF-CNN 的参数列在表 II 中，所提出的基于 SF-CNN 方法的计算复杂度在$10^8$数量级，而 MMSE 需要更高的复杂度在$10^9$数量级。此外，使用 NVIDIA GeForce GTX 1080 Ti GPU, SF-CNN 能够以更高效的并行方式运行，对于一个信道的预测的运行时间仅为$1.47 \times 10^{-4}$秒。相比之下，MMSE 在 Intel(R) Core(TM) i7-3770 CPU 上单信道预测消耗大约$6.14 \times 10^{-2}$秒的时间。

# 4.基于 SFT-CNN 和 SPR-CNN 的信道估计

在本节中，我们首先开发了基于 SFT-CNN 的信道估计方法，该方法进一步将信道时间相关性纳入到 SF-CNN 中。然后将 SFT-CNN 修改为 SPR-CNN，以缓解大规模天线阵列引起的巨大空间导频开销。

对于时变信道，(4)中第 k 个子载波处的频域信道为[7]

$$\mathbf{H}_k(t)=\sqrt{\frac{N_{\mathrm{T}}N_{\mathrm{R}}}{L}}\sum_{l=1}^{L}\alpha_l e^{-j2\pi\left(\tau_l f_s\frac{k}{K}-\nu_l t\right)}\mathbf{a}_{\mathrm{R}}(\varphi_l)\mathbf{a}_{\mathrm{T}}^{H}(\phi_l), \tag{14}$$

其中$\nu_l$为第 l 条路径的多普勒频移。

根据[26]和[27]，信道在连续相干间隔内的时间相关性可以用高斯-马尔可夫分布来建模

$$\mathbf{H}_k[n]=\rho\mathbf{H}_k[n-1]+\sqrt{1-\rho^2}\Theta[n], n\in\mathbb{N}_+, \tag{15}$$

其中$\mathbf{H}_k[n]=\mathbf{H}_k(nT)$为$\mathbf{H}_k(t)$的离散时间版本，$T$表示相干区间的长度，$0\le\rho\le1$为时间相关系数，而$\Theta[n]$为表示每个条目单位方差的创新过程的随机矩阵。(15)清楚地表明，从先前的相干间隔到当前的相干间隔存在着某种固有的信道变化，这种相关性除了空间和频率相关性外，还可以用来提高信道估计精度。接下来，我们首先阐述基于 SFT-CNN 的信道估计。

## 4.1.基于 SFT-CNN 的信道估计方法

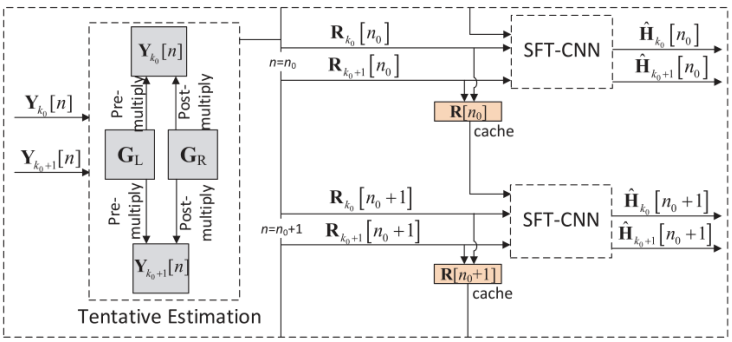



Fig. 2. SFT-CNN based channel estimation.

如图 2 所示，为了便于说明，我们仍然考虑$Q(=2)$相邻子载波$k_0$和$k_0+1$的信道估计。在时变信道中，在(6)中接收机处合并后的接收导频为

$$\mathbf{Y}_k[n]=\sqrt{P}\mathbf{W}^H\mathbf{H}_k[n]\mathbf{F}+\widetilde{\mathbf{N}}_k[n], k\in\{k_0,k_0+1\} \tag{16}$$

类似于基于 SF-CNN 的信道估计，$\mathbf{Y}_k[n]$随后由 TE 模块进行处理，生成初步

估计的信道矩阵，依次为

$$\mathbf{R}_k[n] = \sqrt{P}\mathbf{G}_{\mathrm{L}}\mathbf{W}^H\mathbf{H}_k[n]\mathbf{F}\mathbf{G}_{\mathrm{R}} + \mathbf{G}_{\mathrm{L}}\tilde{\mathbf{N}}_k[n]\mathbf{G}_{\mathrm{R}} \tag{17}$$

然后，SFT-CNN 通过同时利用信道的空间、频率和时间相关性进一步细化这些初步估计的信道矩阵。如图 2 所示，我们捕捉$\mathrm{S}(=2)$连续相干间隔$\mathrm{n}_0$和$(\mathrm{n}_0+1)$来描述信道估计过程。在第$\mathrm{n}_0$个相干区间内，将初步估计的信道矩阵$\mathbf{R}_{k_0}[\mathrm{n}_0]$和$\mathbf{R}_{k_0+1}[\mathrm{n}_0]$输入 SFT-CNN。$\mathbf{R}_{k_0}[\mathrm{n}_0]$和$\mathbf{R}_{k_0+1}[\mathrm{n}_0]$的副本存储在缓存中，以便在下一个相干间隔中使用。在第$(\mathrm{n}_0+1)$相干区间内，SFT-CNN 接收初步估计的信道矩阵$\mathbf{R}_{k_0}[\mathrm{n}_0+1]$和$\mathbf{R}_{k_0+1}[\mathrm{n}_0+1]$，然后从缓存中取$\mathbf{R}_{k_0}[\mathrm{n}_0]$和$\mathbf{R}_{k_0+1}[\mathrm{n}_0]$进行联合处理，得到估计的信道矩阵为

$$\begin{aligned}&\{\widehat{\mathbf{H}}_{k_0}[n_0+1], \widehat{\mathbf{H}}_{k_0+1}[n_0+1]\}\\&= f_\Psi\big(\mathbf{R}_{k_0}[n_0], \mathbf{R}_{k_0+1}[n_0], \mathbf{R}_{k_0}[n_0+1], \mathbf{R}_{k_0+1}[n_0+1]; \Psi\big)\end{aligned} \tag{18}$$

其中$\Psi$表示 SFT-CNN 的参数集。同时，通过将$\mathbf{R}_{k_0}[\mathrm{n}_0]$和$\mathbf{R}_{k_0+1}[\mathrm{n}_0]$替换为$\mathbf{R}_{k_0}[\mathrm{n}_0+1]$和$\mathbf{R}_{k_0+1}[\mathrm{n}_0+1]$从而更新缓存。在每个相干间隔中，使用相同的 SFT-CNN，因为它学习了一般的信道时间相关性，而不是两个连续相干间隔中信道之间的具体关系。

在总结了频道估计过程之后，我们着重介绍了 SFT-CNN 的离线训练。与 SFCNN 类似，在仿真环境中根据一定的信道模型生成由$N_{tr}$个样例组成的训练集，$(\underline{\mathbf{R}}_i, \underline{\mathbf{H}}_i)$表示第 i 个样本。$\underline{\mathbf{R}}_i \in \mathbb{C}^{N_R \times N_T \times 4}$是三维矩阵，包含了在第$\mathrm{n}_0'$和第$\mathrm{n}_0'+1$相干间隔内由 (17) 获得的初步估计信道矩阵，即为$\underline{\mathbf{R}}_i = \left[\mathbf{R}^i_{k_0'}[n_0'], \mathbf{R}^i_{k_0'+1}[n_0'], \mathbf{R}^i_{k_0'}[n_0'+1], \mathbf{R}^i_{k_0'+1}[n_0'+1]\right]$，$\mathrm{n}_0' \in \mathbb{N}_+$。$\underline{\mathbf{H}}_i \in \mathbb{C}^{N_R \times N_T \times 2}$也是在第$\mathrm{n}_0'+1$相干间隔内一个由缩放后的真实信道矩阵组成的，即$\underline{\mathbf{H}}_i = \left[\frac{\mathbf{H}^i_{k_0'}[n_0'+1]}{c}, \frac{\mathbf{H}^i_{k_0'+1}[n_0'+1]}{c}\right]$。与前面一样，$\mathrm{c}>0$是缩放常数，使所有目标数据，$\underline{\mathbf{H}}_i$，的实部和虚部的取值范围与 SFT-CNN 输出层中应用的激活函数相匹配。然后将$\underline{\mathbf{R}}_i$输入 SFT-CNN，逼近相应的缩放真信道$\underline{\mathbf{H}}_i$。SFT-CNN 的结构类似于 SF-CNN，除了它有来自先前相干区间的额外输入。通过估计的缩放信道矩阵，

$\frac{\mathbf{H}^i_{k_0'}[n_0'+1]}{c}, \frac{\mathbf{H}^i_{k_0'+1}[n_0'+1]}{c}$,SFT-CNN 离线训练的目标是最小化 MSE 损失函数

$$\begin{aligned}\mathrm{MSE}_{\mathrm{Loss}} = \frac{1}{N_{\mathrm{tr}}c^2}\sum_{i=1}^{N_{\mathrm{ur}}}\sum_{q=1}^{2} \| \mathbf{H}^i_{k_0'+q-1}[n_0'+1] \\ -\widehat{\mathbf{H}}^i_{k_0'+q-1}[n_0'+1] \|_F^2.\end{aligned} \tag{19}$$

与 SF-CNN 相比，SFT-CNN 仅增加了第一卷积层的计算复杂度$S$倍，根据(12)和表 II 可知，这在总计算复杂度中只是很小的一部分。相比之下，如果进一步考虑时间相关性，则(13)中 MMSE 信道估计的复杂度变得更大

$$C_{\mathrm{MMSE}} \sim \mathcal{O}(S^3Q^3N_{\mathrm{T}}^3N_{\mathrm{R}}^3) \tag{20}$$

即使$S=2$，也会显著增加。因此，SFT-CNN 提供了一种同时利用信道空间、频率和时间相关性来提高信道估计精度的简单而有效的方法。

## 4.2.SFT-CNN 还是 LSTM-CNN？

当我们考虑时间相关性时，LSTM-CNN 自然会出现在我们的脑海中。在本节中，我们将比较 SFT-CNN 和 LSTM-CNN 在估计精度和复杂度方面的差异，这就解释了为什么使用 SFT-CNN 而不是 LSTM-CNN。

1）*架构*：LSTM-CNN 由 1 个输入层、3 个卷积 LSTM 层、7 个卷积层和 1 个输出层组成，其中每个卷积 LSTM 层的处理遵循[28]中的操作。LSTM-CNN 的详细架构如表 3 所示。

TABLE III
ARCHITECTURE OF THE LSTM-CNN

| Layer | Type | Number of filters | Size of filters | Activation function | Recurrent activation function |
|---|---|---|---|---|---|
| 1 | Input | - | - | - | - |
| 2∼4 | Conv. LSTM | 4 | $3\times3$ | tanh | tanh |
| 5∼11 | Conv. | 64 | $3\times3$ | ReLU | - |
| 12 | Output | $2Q$ | $3\times3$ | tanh | - |

2）*信道估计的准确性*：为了测量信道估计性能，我们使用归一化 MSE (NMSE)定义为

$$\mathrm{NMSE} = \mathbb{E}_{\mathbf{H}}\{\| \mathbf{H}-\widehat{\mathbf{H}} \|_F^2 / \| \mathbf{H} \|_F^2\}, \tag{21}$$

其中$\mathbf{H}$和$\widehat{\mathbf{H}}$分别为真实和估计的信道。图 3 给出了基于 SFT-CNN 和基于 LSTM-CNN 的信道估计的 NMSE 性能，其中分别涉及两个和四个相干间隔。从图中可以看出，SFT-CNN 和 LSTM-CNN 在整个信噪比(SNR)范围内取得了几乎相同的性能。

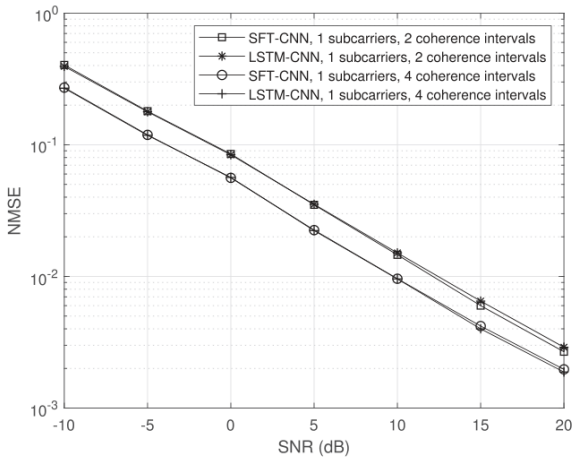



Fig. 3. NMSE for SFT-CNN and LSTM-CNN based channel estimation.

3）*训练和测试复杂性*:在相同的 NMSE 性能下，复杂性成为方法选择的一个重要指标。表 4 给出了 SFT-CNN 和 LSTM-CNN 在 NVIDIA GeForce GTX 1080 Ti GPU 上训练和测试的时间复杂度，其中$N_C$表示 CNN 所涉及的相干间隔数。很明显，LSTM-CNN 在训练和测试阶段都比 SFT-CNN 花费更多的时间。

TABLE IV
TIME COMPLEXITY OF TRAINING AND TESTING

| | Training time seconds/epoch | | Testing time seconds/channel realization | |
|---|---|---|---|---|
| | $N_c = 2$ | $N_c = 4$ | $N_c = 2$ | $N_c = 4$ |
| SFT-CNN | 22 | 23 | $1.51 \times 10^{-4}$ | $1.7 \times 10^{-4}$ |
| LSTM-CNN | 34 | 48 | $3.53 \times 10^{-4}$ | $5.14 \times 10^{-4}$ |

## 4.3.基于 SPR-CNN 的信道估计

发射端和接收端的大规模阵列天线在空间域上产生巨大的导频开销。在这个小节，我们设计了基于 SPR-CNN 的信道估计，使用较少的导频，但仍然保证了较好的准确性。

基于 SPR-CNN 的信道估计的基本思想可以总结如下:

1）将$D$个连续相干间隔分组作为 CEU，利用信道相关性降低空间导频开销。不同 CEU 不重叠。

2）在每个 CEU 的第一个相干区间内，使用全空间导频开销进行信道估计。[2]然后在随后的相干间隔内减小导频开销。

3）对于第一个相干间隔，接收端使用当前接收的导频估计当前信道。对于其余的相干间隔，接收器使用当前和之前在该 CEU 中接收到的所有导频来联

[2] 全空间导频开销是指波束形成向量的数量等于发射天线的数量，组合向量的数量等于接收天线的数量。

合估计当前信道。

下面是详细的信道估计程序。每个 CEU 的相干间隔采用不同的波束形成和组合矩阵。如图 4 所示，在(16)的接收机处合并后的接收导频为

$$\mathbf{Y}_k[n] = \sqrt{P}\mathbf{W}^H[n]\mathbf{H}_k[n]\mathbf{F}[n] + \widetilde{\mathbf{N}}_k[n] \tag{22}$$

其中$\mathbf{F}[n] \in \mathbb{C}^{N_T \times M_T[n]}$以及$\mathbf{F}[n] \in \mathbb{C}^{N_R \times M_R[n]}$分别表示第 n 个相干区间内的波束形成矩阵和组合矩阵。相应的空间引航开销为

$$p[n] = M_{\mathrm{T}}[n]\left\lceil \frac{M_{\mathrm{R}}[n]}{N_{\mathrm{R}}^{\mathrm{RF}}} \right\rceil \tag{23}$$

从(23)可以通过降低$M_T[n]$或/和$M_{\mathrm{R}}[n]$来节省空间导频开销。

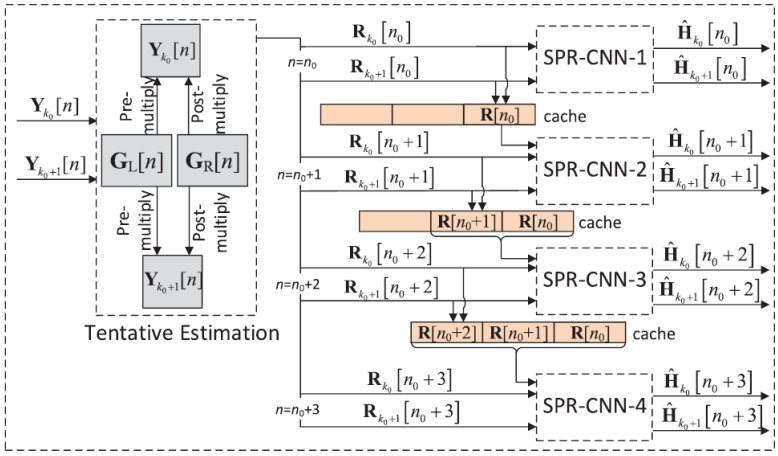



Fig. 4. SPR-CNN-based channel estimation.

首先用 TE 模块处理$\mathbf{Y}_k[n]$，并给出初步估计的信道矩阵

$$\begin{aligned}\mathbf{R}_k[n] &= \mathbf{G}_{\mathrm{L}}[n]\mathbf{Y}_k[n]\mathbf{G}_{\mathrm{R}}[n] \\ &= \sqrt{P}\mathbf{G}_{\mathrm{L}}[n]\mathbf{W}^H[n]\mathbf{H}_k[n]\mathbf{F}[n]\mathbf{G}_{\mathrm{R}}[n] \\ &\quad +\mathbf{G}_{\mathrm{L}}[n]\widetilde{\mathbf{N}}_k[n]\mathbf{G}_{\mathrm{R}}[n]\end{aligned} \tag{24}$$

其中$\mathbf{G}_{\mathrm{L}}[n]$和$\mathbf{G}_{\mathrm{R}}[n]$也随着相干间隔指数的变化而变化，表示为

$$\mathbf{G}_L[\mathrm{n}] = \begin{cases} \mathbf{W}[\mathrm{n}]^H, & M_R[\mathrm{n}] < N_R[\mathrm{n}] \\ (\mathbf{W}[\mathrm{n}]\mathbf{W}^H[\mathrm{n}])^{-1}\mathbf{W}[\mathrm{n}], & M_R[\mathrm{n}] \geq N_R[\mathrm{n}] \end{cases} \tag{25}$$

且

$$\mathbf{G}_R[\mathrm{n}] = \begin{cases} \mathbf{F}^H[\mathrm{n}], & M_T[\mathrm{n}] < N_T[\mathrm{n}] \\ \mathbf{F}^H[\mathrm{n}](\mathbf{F}[\mathrm{n}]\mathbf{F}^H[\mathrm{n}])^{-1}, & M_T[\mathrm{n}] \geq N_T[\mathrm{n}] \end{cases} \tag{26}$$

然后$\mathbf{R}_k[n]$由 CNN 估计部分进行处理。我们将每$D(=4)$个连续的相干间隔分组为一个 CEU，并从第$\mathrm{n}_0$到$(\mathrm{n}_0+3)$个相干间隔捕获一个 CEU，如图 4 所示。SFT-CNN 共有 4 个，输入和输出不同，分别称为 SPR-CNN-1、2、3、4。在第$\mathrm{n}_0$

个相干区间内，全空间导频开销，即$M_T[\mathrm{n}] = N_T$和$M_R[\mathrm{n}] = N_\mathrm{R}$，旨在提供该 CEU 所有相干区间的精确信道信息。在 TE 模块之后，将$\mathbf{R}_{k_0}[\mathrm{n}_0]$和$\mathbf{R}_{k_0+1}[\mathrm{n}_0]$输入到 SPR-CNN-1 中，生成最终估计的信道矩阵$\widehat{\mathbf{H}}_{k_0}[\mathrm{n}_0]$和$\widehat{\mathbf{H}}_{k_0+1}[\mathrm{n}_0]$。同时，在缓存中存储$\mathbf{R}_{k_0}[\mathrm{n}_0]$和$\mathbf{R}_{k_0+1}[\mathrm{n}_0]$的副本，为后续相干间隔的信道估计提供额外的信道信息。在$(\mathrm{n}_0+1)$相干区间内，为了节省导频开销，将$\mathbf{F}[\mathrm{n}]$和$\mathbf{W}[\mathrm{n}]$的维度减小，即$M_T[\mathrm{n}] < N_T$和$M_R[\mathrm{n}] < N_\mathrm{R}$。[3] $\mathbf{R}_{k_0}[\mathrm{n}_0]$和$\mathbf{R}_{k_0+1}[\mathrm{n}_0]$以及$\mathbf{R}_{k_0}[\mathrm{n}_0+1]$和$\mathbf{R}_{k_0+1}[\mathrm{n}_0+1]$同时输入到 SPR-CNN-2 中，得到$\widehat{\mathbf{H}}_{k_0}[\mathrm{n}_0+1]$和$\widehat{\mathbf{H}}_{k_0+1}[\mathrm{n}_0+1]$。除了$\mathbf{R}_{k_0}[\mathrm{n}_0]$和$\mathbf{R}_{k_0+1}[\mathrm{n}_0]$外，缓存中还存储了$\mathbf{R}_{k_0}[\mathrm{n}_0+1]$和$\mathbf{R}_{k_0+1}[\mathrm{n}_0+1]$。利用缓存中存储的所有矩阵对第$(\mathrm{n}_0+2)$个相干区间进行联合信道估计。$(\mathrm{n}_0+2)$和$(\mathrm{n}_0+3)$相干区间的信道估计与$(\mathrm{n}_0+1)$相干区间的信道估计相似。在第$(\mathrm{n}_0+3)$相干区间内进行信道估计后，缓存将被清空，然后用于下一个 CEU。从图 4 中，四个不同的 SPR-CNN 分别用于 CEU 的相干间隔，并在所有 CEU 中重用。SPR-CNN 的结构和训练过程与 SFT-CNN 相似，只是不同 SPR-CNN 的输入矩阵数量不同。算法 1 给出了基于 SPR-CNN 的信道估计的直观描述。

在 4 个 SPR-CNN 中，SPR-CNN-4 的复杂度最高，输入矩阵最多。但与第三节的 SF-CNN 相比，它只是增加了第一卷积层$D(=4)$倍的复杂度，对总计算复杂度的影响有限。因此，基于 SPR-CNN 的信道估计在适度增加复杂度的同时，有效地节省了空间导频开销。

我们设计了基于 SPR-CNN 的信道估计，目的是在显著降低空间域导频开销的同时，仍然保证较好的信道估计精度。在每个有$D$个相干间隔的 CEU 中，在第一个相干间隔中使用完整的导频开销，而在其余的相干间隔中使用减少的导频开销。第一个相干区间提供了完整的信道信息，这对该 CEU 中所有相干区间的信道估计很有帮助。很明显，CEU 的平均导频开销随着$D$的增加而减少。但是，增加$D$也会降低平均信道估计精度，因为第一个相干区间提供的完整信道信息的影响随着时间相关性的消失而减弱。因此，我们的 SPR-CNN 架构在一个 CEU 中包含了四个相干间隔，以在导频开销和估计精度之间实现良好的权衡。另外，如

[3] 在一个 CEU 的不同相干间隔中可以使用不同的导频开销。但是对于每个相干间隔，它和它在其他 CEU 中的对应部分应该使用相同的导频开销。

果使用 LSTM-CNN 进行图 4 中的信道估计，则需要 4 个结构如表 3 所示的 LSTM-CNN。如第四-2 节所述，LSTM-CNN 在性能上并不优于简单的 SFT-CNN 架构，但其复杂性要高得多，不适合基于 SPR-CNN 的信道估计。

**Algorithm 1:** SPR-CNN based Channel Estimation.

**Input:** The total number of CEUs $M_{\text{CEU}}$, the number of coherence intervals in each CEU $D$, spatial pilot overhead from the second to $D$th coherence intervals of each CEU
**Output:** Estimated channel matrices
**Procedure:**

Initialize the CEU and coherence interval indices as $m = 1$ and $d = 1$;
Train $D$ different SPR-CNNs for the first to $D$th coherence intervals;
**for** $m \in [1, M_{\text{CEU}}]$ **do**
  **for** $d \in [1, D]$ **do**
    **if** $d = 1$ **then**
      Use full spatial pilot overhead;
      Tentatively estimate channels according to (24);
      Store the tentatively estimated channel matrices in the cache;
      Input the tentatively estimated channel matrices of the first coherence interval into SPR-CNN-1 to obtain the estimated channel matrices of the first coherence interval;
    **else**
      Use reduced spatial pilot overhead;
      Tentatively estimate channels according to (24);
      Store the tentatively estimated channel matrices in the cache (Invalid for $d = D$);
      Input the tentatively estimated channel matrices from the first to $d$th coherence intervals into SPR-CNN-$d$ to obtain the estimated channel matrices of the $d$th coherence interval;
    **end if**
  **end for**
  Empty the cache and reset $d = 1$
**end for**
**return** the estimated channel matrices of $M_{\text{CEU}}$ CEUs

# 5.数值结果

略

# 6.结论

在本文中，我们开发了基于深度 CNN 的 mmWave 大规模 MIMO-OFDM 系统的信道估计方法。首先提出了基于 SF-CNN 的信道估计方法，同时利用空间相关和频率相关。为了进一步在真实场景中纳入时间相关性，我们开发了基于 SFT-CNN 的方法。最后，考虑到大量天线造成的巨大空间导频开销，我们设计了基于 SPR-CNN 的信道估计方法来缓解这一问题。数值结果表明，基于 SF-CNN 和

基于 SFT-CNN 的方法，复杂度较低，性能优于非理想 MMSE 估计器，且非常接近理想 MMSE 估计器。即使信道统计值不同，所提出的方法仍然可以获得相当好的性能。基于 SPR-CNN 的信道估计方法在性能损失较小的情况下，能有效地节省空间导频开销。

所提出的基于 CNN 的信道估计方法在对尽可能多的信道统计量的鲁棒性方面可以进一步改进。为了实现这一目标，需要对复杂的 mmWave MIMO 信道结构进行深入剖析，为 DNN 结构设计和超参数调谐提供见解。

# 7.参考文献

[1] P. Dong, H. Zhang, G. Y. Li, N. NaderiAlizadeh, and I. Gaspar, “Deep CNN for wideband mmWave massive MIMO channel estimation using frequency correlation,” in Proc. IEEE Int. Conf. Acoust., Speech, Signal Process., Brighton, U.K., May 2019, pp. 4529–4533.
[2] A. L. Swindlehurst, E.Ayanoglu, P. Heydari, and F. Capolino, “Millimeterwave massive MIMO: The next wireless revolution?” IEEE Commun. Mag., vol. 52, no. 9, pp. 56–62, Sep. 2014.
[3] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, “Massive MIMO for next generation wireless systems,” IEEE Commun. Mag., vol. 52, no. 2, pp. 186–195, Feb. 2014.
[4] H. Q. Ngo, E. G. Larsson, and T. L. Marzetta, “Energy and spectral efficiency of very large multiuser MIMO systems,” IEEE Trans. Commun., vol. 61, no. 4, pp. 1436–1449, Apr. 2013.
[5] B. Wang, F. Gao, S. Jin, H. Lin, and G. Y. Li, “Spatial- and frequency-wideband effects in millimeter-wave massive MIMO systems,” IEEE Trans. Signal Process., vol. 66, no. 13, pp. 3393–3406, Jul. 2018.
[6] M. Jian, F. Gao, Z. Tian, S. Jin, and S. Ma, “Angle-domain aided UL/DL channel estimation for wideband mmWave massive MIMO systems with beam squint,” IEEE Trans. Wireless Commun., to be published.
[7] R.W.Heath Jr., N. Gonzlez-Prelcic, S. Rangan,W.Roh, andA.M. Sayeed, “An overview of signal processing techniques for millimeter wave MIMO systems,” IEEE J. Sel. Topics in Signal Process., vol. 10, no. 3, pp. 436–453, Apr. 2016.
[8] L. Liang, W. Xu, and X. Dong, “Low-complexity hybrid precoding in massive multiuserMIMO systems,” IEEE Wireless Commun. Lett., vol. 3, no. 6, pp. 653–656, Dec. 2014.
[9] A. Alkhateeb, O. E. Ayach, G. Leus, and R. W. Heath Jr., “Channel estimation and hybrid precoding for millimeter wave cellular systems,” IEEE J. Sel. Topics Signal Process., vol. 8, no. 5, pp. 831–846, Oct. 2014.
[10] Z. Gao, C. Hu, L. Dai, and Z. Wang, “Channel estimation for millimeterwave massive MIMO with hybrid precoding over frequency-selective fading channels,”IEEE Commun. Lett., vol. 20, no. 6, pp. 1259–1262, Jun. 2016.
[11] Y . Wang, W. Xu, H. Zhang, and X. Y ou, “Wideband mmWave channel estimation for hybrid massive MIMO with low-precision ADCs,”IEEE Wireless Commun. Lett., vol. 8, no. 1, pp. 285–288, Feb. 2019.
[12] Z. Zhou, J. Fang, L. Y ang, H. Li, Z. Chen, and R. S. Blum, “Low-rank tensor decomposition-aided channel estimation for millimeter wave MIMOOFDM systems,”IEEE J. Sel. Areas Commun., vol. 35, no. 7, pp. 1524– 1538, Jul. 2017.
[13] H. Ghauch, T. Kim, M. Bengtsson, and M. Skoglund, “Subspace estimation and decomposition for large millimeter-wave MIMO systems,”IEEE J. Sel. Topics Signal Process., vol. 10, no. 3, pp. 528–542, Apr. 2016.
[14] Z. Qin, H. Y e, G. Y . Li, and B.-H. Juang, “Deep learning in physical layer communications,”IEEE Wireless Commun., vol. 26, no. 2, pp. 93–99, Apr. 2019.

[15] H. Y e, G. Y . Li, and B.-H. Juang, “Power of deep learning for channel estimation and signal detection in OFDM systems,”IEEE Wireless Commun. Lett., vol. 7, no. 1, pp. 114–117, Feb. 2018.
[16] H. He, C.-K. Wen, S. Jin, and G. Y . Li, “Deep learning-based channel estimation for beamspace mmWave massive MIMO systems,”IEEE Wireless Commun. Lett., vol. 7, no. 5, pp. 852–855, Oct. 2018.
[17] C.-K. Wen, W.-T. Shih, and S. Jin, “Deep learning for massive MIMO CSI feedback,”IEEE Wireless Commun. Lett., vol. 7, no. 5, pp. 748–751, Oct. 2018.
[18] T. Wang, C.-K. Wen, S. Jin, and G. Y . Li, “Deep learning-based CSI feedback approach for time-varying massive MIMO channels,”IEEE Wireless Commun. Lett., vol. 8, no. 2, pp. 416–419, Apr. 2019.
[19] H. He, S. Jin, C.-K. Wen, F. Gao, G. Y . Li, and Z. Xu, “Model-driven deep learning for physical layer communications,”IEEE Wireless Commun., t o be published.
[20] J.-M. Kang, C.-J. Chun, and I.-M. Kim, “Deep-learning-based channel estimation for wireless energy transfer,”IEEE Commun. Lett., vol. 22, no. 11, pp. 2310–2313, Nov. 2018.
[21] P . Dong, H. Zhang, and G. Y . Li, “Machine learning prediction based CSI acquisition for FDD massive MIMO downlink,” inProc. IEEE Global Commun. Conf., Abu Dhabi, United Arab Emirates, Dec. 2018, pp. 1–6.
[22] Y .-S. Jeon, S.-N. Hong, and N. Lee, “Blind detection for MIMO systems with low-resolution ADCs using supervised learning,” inProc. IEEE Int. Conf. Commun., Paris, France, May 2017, pp. 1–6.
[23] K. Simonyan and A. Zisserman, “V ery deep convolutional networks for large-scale image recognition,”arXiv: 1409.1556, 2014.
[24] R. Hunger, “Floating point operations in matrix-vector calculus,” Technische Universität München, Associate Institute for Signal Processing, Tech. Rep. v1.3, 2007.
[25] K. He and J. Sun, “Convolutional neural networks at constrained time cost,” inProc. IEEE Conf. Comput. Vision Pattern Recognit., Boston, MA, USA, Jun. 2015, pp. 5353–5360.
[26] V . V a, J. Choi, and R. W. Heath Jr., “The impact of beamwidth on temporal channel variation in vehicular channels and its implications,” IEEE Trans. V eh. Technol., vol. 66, no. 6, pp. 5014–5029, Jun. 2017.
[27] J. Choi, D. Love, and P . Bidigare, “Downlink training techniques for FDD massive MIMO systems: Open-loop and closed-loop training with memory,”IEEE J. Sel. Topics Signal Process., vol. 8, no. 5, pp. 802–814, Oct. 2014.
[28] X. Shi, Z. Chen, H. Wang, D. Y . Y eung, W. K. Wong, and W. C. Woo, “Convolutional LSTM network: A machine learning approach for precipitation nowcasting,” inProc. Neural Inf. Process. Syst., 2015, pp. 802–810.
[29] 3GPP , “5G; Study on channel model for frequencies from 0.5 to 100 GHz,” 3rd Generation Partnership Project (3GPP), Tech. Rep. 38.901 V15.0.0, Jun. 2018.

# 四、外文原文

# Deep CNN-Based Channel Estimation for mmWave Massive MIMO Systems

Peihao Dong, *Student Member, IEEE*, Hua Zhang, *Member, IEEE*, Geoffrey Ye Li, *Fellow, IEEE*, Ivan Simões Gaspar, and Navid NaderiAlizadeh

***Abstract*—For millimeter wave (mmWave) massive multiple-input multiple-output (MIMO) systems, hybrid processing architecture is usually used to reduce the complexity and cost, which poses a very challenging issue in channel estimation. In this paper, deep convolutional neural network (CNN) is employed to address this problem. We first propose a spatial-frequency CNN (SF-CNN) based channel estimation exploiting both the spatial and frequency correlation, where the corrupted channel matrices at adjacent subcarriers are input into the CNN simultaneously. Then, exploiting the temporal correlation in time-varying channels, a spatial-frequency-temporal CNN (SFT-CNN) based approach is developed to further improve the accuracy. Moreover, we design a spatial pilot-reduced CNN (SPR-CNN) to save spatial pilot overhead for channel estimation, where channels in several successive coherence intervals are grouped and estimated by a channel estimation unit with memory. Numerical results show that the proposed SF-CNN and SFT-CNN based approaches outperform the non-ideal minimum mean-squared error (MMSE) estimator but with reduced complexity, and achieve the performance close to the ideal MMSE estimator that is very difficult to be implemented in practical situations. They are also robust to different propagation scenarios. The SPR-CNN based approach achieves comparable performance to SF-CNN and SFT-CNN based approaches while only requires about one-third of spatial pilot overhead at the cost of complexity. The results in this paper clearly demonstrate that deep CNN can efficiently exploit channel correlation to improve the estimation performance for mmWave massive MIMO systems.**

***Index Terms*—mmWave massive MIMO, deep CNN, channel estimation, channel correlation.**

Manuscript received February 15, 2019; revised May 26, 2019; accepted June 19, 2019. Date of publication July 1, 2019; date of current version September 20, 2019. The work of P. Dong was supported in part by the National Key R&D Program of China under Grant 2018YFB2202200, in part by the National Natural Science Foundation of China under Grant 61571118, and in part by the China Scholarship Council under Grant 201706090059. The work of H. Zhang was supported in part by the National Key R&D Program of China under Grant 2018YFB2202200 and in part by the National Natural Science Foundation of China under Grant 61571118. The work of G. Y. Li was supported in part by a research gift from Intel Corporation and in part by National Science Foundation under Grants 1731017 and 1815637. The guest editor coordinating the review of this paper and approving it for publication was Prof. Feifei Gao. *(Corresponding author: Hua Zhang.)*

P. Dong and H. Zhang are with National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China (e-mail: phdong@seu.edu.cn; huazhang@seu.edu.cn).

G. Y. Li is with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: liye@ece.gatech.edu).

I. S. Gaspar and N. NaderiAlizadeh are with Intel Corporation, Santa Clara, CA 95054 USA (e-mail: ivan.simoes.gaspar@intel.com; navid.naderializadeh@intel.com).

Digital Object Identifier 10.1109/JSTSP.2019.2925975

## I. Introduction

MILLIMETER wave (mmWave) communications can meet the high data rate demand due to its large bandwidth. Its high propagation loss can be well compensated by using massive multiple-input multiple-output (MIMO) technique [2]−[6]. However, Due to the limited physical space with closely placed antennas and prohibitive power consumption in mmWave massive MIMO systems, it is difficult to equip a dedicated radio frequency (RF) chain for each antenna. To reduce complexity and cost, phase shifter based two-stage architecture, usually called hybrid architecture, is widely used at both the transmitter and the receiver to connect a large number of antennas with much fewer RF chains [7], [8].

For mmWave massive MIMO systems with the hybrid architecture, channel estimation is a challenging problem. In [9], a hierarchical multi-resolution codebook has been designed, based on which an adaptive channel estimation algorithm has been developed by exploiting the channel sparsity. In [10], the structured sparsity in angle domain has been utilized to estimate the wideband channel for multi-user mmWave massive MIMO uplink. In [11], a channel estimation approach has been developed for mmWave massive MIMO orthogonal frequency division multiplexing (OFDM) systems with low-precision analog-to-digital converters (ADCs). For the mmWave MIMO-OFDM downlink, a channel parameter estimation for the angles, time delays, and fading coefficients has been proposed in [12] resorting to the low-rank tensor decomposition. Instead of estimating the mmWave MIMO channel directly, the method for singular subspace estimation has been proposed in [13], based on which a subspace decomposition algorithm has been further developed to design the hybrid analog-digital architecture.

Compared to the conventional methods, machine learning (ML) is more powerful to uncover the inherent characteristics inside data/signals collected in an end-to-end manner and thus can achieve better performance when addressing various problems in wireless communications [14]. In [15], deep learning (DL) has been successfully used in joint channel estimation and signal detection of OFDM systems with interference and nonlinear distortions. In [16], iterative channel estimation has been proposed for the lens mmWave massive MIMO systems, where denoising neural network (NN) is used in each iteration to update the estimated channel. To reduce the CSI feedback overhead of the frequency duplex division (FDD) massive MIMO system, DL has been employed in [17] to compress the channel into a low

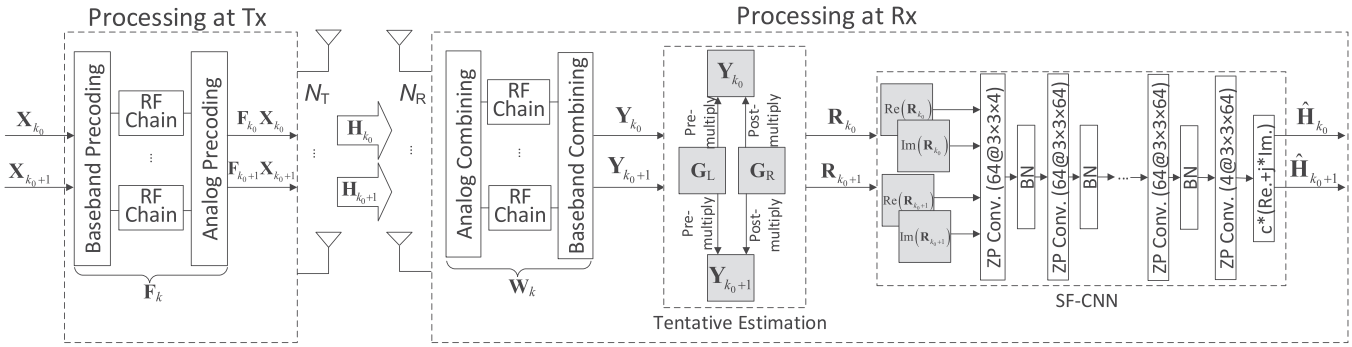



Fig. 1. SF-CNN based channel estimation.

dimensional codeword and then to perform recovery with high accuracy. Exploiting temporal correlation of the channel, long short-term memory (LSTM) based deep NN (DNN) has been introduced in [18] to develop a more efficient channel compression and recovery method for the CSI feedback. More related research can be also found in [19] and the references therein. In [20], DL has been applied to estimate channels in wireless power transfer systems, which outperforms the conventional scheme in terms of both estimation accuracy and harvested energy. In [21], supervised learning algorithms have been used to acquire the downlink CSI for FDD massive MIMO systems with reduced overheads for pilot and CSI feedback. In [22], the supervised learning has been exploited to perform blind detection for massive MIMO systems with low-precision ADCs.

The conventional channel estimation methods usually perform unsatisfactorily in the more practical complicated channel model and also suffer from high complexity. In contrast, the deep convolutional NN (CNN) is more capable to extract the inherent characteristics underlying the channel matrix from the large amount of data and provides the potential to estimate the channel more accurately with lower complexity by using the efficient parallel computing methods. In this paper, we use the deep CNN to address channel estimation for mmWave massive MIMO-OFDM systems. To exploit the correlation among channels at adjacent subcarriers in OFDM, we first propose a spatial-frequency CNN (SF-CNN) based channel estimation, where the tentatively estimated channel matrices at adjacent subcarriers are input into the CNN simultaneously [1]. To further exploit the temporal correlation, a spatial-frequency-temporal CNN (SFT-CNN) based channel estimation is developed, where the channel information of the previous coherence interval is utilized when estimating the channel matrices of the current coherence interval. The SFT-CNN based approach incorporates all types of channel correlation in a simple way and yields remarkable performance gains that can be used to significantly save the spatial pilot overhead due to large-scale arrays. Therefore, we propose a spatial pilot-reduced CNN (SPR-CNN) based channel estimation, where channels in several successive coherence intervals are grouped and estimated by a channel estimation unit (CEU) with memory. From the numerical results, the proposed SF-CNN and SFT-CNN based approaches outperform the non-ideal minimum mean-squared error (MMSE) estimator and achieve the performance very close to the ideal MMSE estimator that is very difficult to be implemented in practical systems. They are also with lower complexity than the MMSE estimator and exhibit the unique robustness to maintain the fairly good performance when facing different channel statistics. The SPR-CNN based approach achieves comparable performance to SF-CNN and SFT-CNN based approaches by using only about one third of spatial pilot overhead and moderately increased complexity.

The rest of the paper is organized as follows. Section II describes the considered mmWave massive MIMO system, followed by the proposed SF-CNN based channel estimation in Section III. Section IV further develops the SFT-CNN and SPR-CNN based channel estimation. Numerical results are provided in Section V and finally Section VI gives concluding remarks.

*Notations*: In this paper, we use upper and lower case boldface letters to denote matrices and vectors, respectively. $\|\cdot\|_F$, $(\cdot)^T$, $(\cdot)^H$, $(\cdot)^{-1}$, and $\mathbb{E}\{\cdot\}$ represent the Frobenius norm, transpose, conjugate transpose, inverse, and expectation, respectively. $\mathcal{CN}(\mu, \sigma^2)$ represents circular symmetric complex Gaussian distribution with mean $\mu$ and variance $\sigma^2$. $\delta(\cdot)$ and $\lceil\cdot\rceil$ denote the delta function and ceiling function. $\mathbb{N}_+$ denotes the set of all positive integers.

## II. System Model

We consider a mmWave massive MIMO-OFDM system as in Fig. 1, where the transmitter is with $N_{\mathrm{T}}$ antennas and $N_{\mathrm{T}}^{\mathrm{RF}}$ RF chains and the receiver is with $N_{\mathrm{R}}$ antennas and $N_{\mathrm{R}}^{\mathrm{RF}}$ RF chains. Phase shifters are employed to connect a large number of antennas with a much fewer number of RF chains at both the transmitter and the receiver sides. We therefore assume $N_{\mathrm{T}} \gg N_{\mathrm{T}}^{\mathrm{RF}}$ and $N_{\mathrm{R}} \gg N_{\mathrm{R}}^{\mathrm{RF}}$.

According to [10], the $N_{\mathrm{R}} \times N_{\mathrm{T}}$ channel matrix between the receiver and the transmitter in the delay domain is given by

$$\mathbf{H}(\tau) = \sqrt{\frac{N_{\mathrm{T}} N_{\mathrm{R}}}{L}} \sum_{l=1}^{L} \alpha_l \delta(\tau - \tau_l) \mathbf{a}_{\mathrm{R}}(\varphi_l) \mathbf{a}_{\mathrm{T}}^H(\phi_l), \quad (1)$$

where $L$ is the number of main paths, $\alpha_l \sim \mathcal{CN}(0, \sigma_\alpha^2)$ is the propagation gain of the $l$th path with $\sigma_\alpha^2$ being the average power gain, $\tau_l$ is the delay of the $l$th path, $\varphi_l$ and $\phi_l \in [0, 2\pi]$ are the azimuth angles of arrival and departure (AoA/AoD) at the receiver and the transmitter, respectively. For uniform linear array

(ULA), the corresponding response vectors can be expressed as

$$\mathbf{a}_{\mathrm{R}}(\varphi_l) = \frac{1}{\sqrt{N_{\mathrm{R}}}}\left[1, e^{-j2\pi\frac{d}{\lambda}\sin(\varphi_l)}, \ldots, e^{-j2\pi\frac{d}{\lambda}(N_{\mathrm{R}}-1)\sin(\varphi_l)}\right]^T, \tag{2}$$

$$\mathbf{a}_{\mathrm{T}}(\phi_l) = \frac{1}{\sqrt{N_{\mathrm{T}}}}\left[1, e^{-j2\pi\frac{d}{\lambda}\sin(\phi_l)}, \ldots, e^{-j2\pi\frac{d}{\lambda}(N_{\mathrm{T}}-1)\sin(\phi_l)}\right]^T, \tag{3}$$

where $d$ and $\lambda$ denote the distance between the adjacent antennas and carrier wavelength, respectively.

According to the channel model in (1), the frequency domain channel of the $k$th subcarrier in OFDM is given by

$$\mathbf{H}_k = \sqrt{\frac{N_{\mathrm{T}}N_{\mathrm{R}}}{L}}\sum_{l=1}^{L}\alpha_l e^{-j2\pi\tau_l f_s\frac{k}{K}}\mathbf{a}_{\mathrm{R}}(\varphi_l)\mathbf{a}_{\mathrm{T}}^H(\phi_l), \tag{4}$$

where $f_s$ denotes the sampling rate and $K$ is the number of OFDM subcarriers.

To estimate $\mathbf{H}_k$, the transmitter activates only one RF chains to transmit the pilot on one beam in each channel use while the receiver combines the received pilot by using all RF chains associated with different beams. In more detail, the transmitter transmits pilots, $x_{k,u}$, using $M_{\mathrm{T}}$ beamforming vectors, $\mathbf{f}_{k,u} \in \mathbb{C}^{N_{\mathrm{T}}\times 1}$, $u = 1, \ldots, M_{\mathrm{T}}$. For the transmit pilot signal corresponding to each beamforming vector, $\mathbf{f}_{k,u}$, the receiver employs $M_{\mathrm{R}}$ combining vectors, $\mathbf{w}_{k,v} \in \mathbb{C}^{N_{\mathrm{R}}\times 1}$, $v = 1, \ldots, M_{\mathrm{R}}$, to respectively process it. Since the receiver is equipped with $N_{\mathrm{R}}^{\mathrm{RF}} (< M_{\mathrm{R}})$ RF chains, it can only use $N_{\mathrm{R}}^{\mathrm{RF}}$ combining vectors in a channel use. Then, if the receiver uses all $M_{\mathrm{R}}$ combining vectors to process a beamforming vector carrying pilot, the required channel uses will be $\lceil\frac{M_{\mathrm{R}}}{N_{\mathrm{R}}^{\mathrm{RF}}}\rceil$. So the total channel uses for processing all beamforming vectors will be $M_{\mathrm{T}}\lceil\frac{M_{\mathrm{R}}}{N_{\mathrm{R}}^{\mathrm{RF}}}\rceil$.[1] Then the pilot signal matrix associated with the $k$th subcarrier at the baseband of the receiver can be written as

$$\mathbf{Y}_k = \mathbf{W}_k^H\mathbf{H}_k\mathbf{F}_k\mathbf{X}_k + \tilde{\mathbf{N}}_k, \tag{5}$$

where $\mathbf{W}_k = [\mathbf{w}_{k,1}, \ldots, \mathbf{w}_{k,M_{\mathrm{R}}}]$ and $\mathbf{F}_k = [\mathbf{f}_{k,1}, \ldots, \mathbf{f}_{k,M_{\mathrm{T}}}]$ are combining matrix and beamforming matrix, respectively, $\mathbf{X}_k$ is an $M_{\mathrm{T}} \times M_{\mathrm{T}}$ diagonal matrix with its $u$th diagonal element being $x_{k,u}$. $\tilde{\mathbf{N}}_k = \mathbf{W}_k^H\mathbf{N}_k$ denotes the effective noise after combining and $\mathbf{N}_k$ is additive white Gaussian noise (AWGN) with $\mathcal{CN}(0,1)$ elements before combining.

We consider the pilot insertion in both frequency and time domain. Specifically, adjacent $Q$ ($Q \geq 2$) subcarriers respectively place pilots with the same length at the beginning of a coherence interval to form a pilot subcarrier block and the rest of time slots in each coherence interval are used for data transmission. Two pilot subcarrier blocks are separated by $Q_{\mathrm{d}}$ ($Q_{\mathrm{d}} \geq 0$) subcarriers dedicated to data transmission. Pilots are utilized to estimate the channels of corresponding time-frequency positions. Based on the estimated channels, interpolation can be applied to get the channels at the positions without pilots. It is clear that the accuracy of interpolation is dependent on the accuracy of the estimated channels and the variation of channels. Therefore, we will focus on improving the accuracy of the pilot based channel estimation in this paper so that more reliable reference values can be provided for interpolation.

[1]This pilot transmission process can capture the main paths in mmWave channels. Although simultaneously activating multiple RF chains with different beams at the transmitter can accelerate the pilot transmission process, it fails to capture the main paths and leads to poor channel estimation performance. Thus in lots of related work, only one RF chain is activated to transmit the pilot on one beam at the transmitter during each channel use to guarantee the performance of channel estimation algorithms [9], [11]. Similarly, in this paper, the pilot transmission mode that we consider also facilitates the CNN based channel estimation to achieve very good accuracy.

## III. SF-CNN-Based Channel Estimation

In this section, we first elaborate the SF-CNN based channel estimation, including an overview of the proposed approach, the offline training of SF-CNN, and the online deployment. Then the computational complexity for the online estimation is analyzed.

### A. Algorithm Description

*1) Channel Estimation Procedure:* Fig. 1 illustrates the channel estimation procedure for adjacent $Q$ ($= 2$) subcarriers, $k_0$ and $k_0 + 1$, to simplify our discussion even if it is trivial to extend to the case with $Q > 2$. Without loss of generality, we assume the worst case that $\mathbf{W}_k = \mathbf{W}$, $\mathbf{F}_k = \mathbf{F}$, and $\mathbf{X}_k = \sqrt{P}\mathbf{I}$ for $k \in \{k_0, k_0 + 1\}$, where $P$ denotes the transmit power. The pilot signal matrix, $\mathbf{Y}_k$, becomes

$$\mathbf{Y}_k = \sqrt{P}\mathbf{W}^H\mathbf{H}_k\mathbf{F} + \tilde{\mathbf{N}}_k. \tag{6}$$

Then $\mathbf{Y}_k$ goes through the tentative estimation (TE) module, which uses two matrices, $\mathbf{G}_{\mathrm{L}}$ and $\mathbf{G}_{\mathrm{R}}$, to process $\mathbf{Y}_k$ and outputs a coarse estimation of $\mathbf{H}_k$, that is,

$$\mathbf{R}_k = \mathbf{G}_{\mathrm{L}}\mathbf{Y}_k\mathbf{G}_{\mathrm{R}} = \sqrt{P}\mathbf{G}_{\mathrm{L}}\mathbf{W}^H\mathbf{H}_k\mathbf{F}\mathbf{G}_{\mathrm{R}} + \mathbf{G}_{\mathrm{L}}\tilde{\mathbf{N}}_k\mathbf{G}_{\mathrm{R}}, \tag{7}$$

where

$$\mathbf{G}_{\mathrm{L}} = \begin{cases}\mathbf{W}, & M_{\mathrm{R}} < N_{\mathrm{R}},\\ (\mathbf{W}\mathbf{W}^H)^{-1}\mathbf{W}, & M_{\mathrm{R}} \geq N_{\mathrm{R}},\end{cases} \tag{8}$$

and

$$\mathbf{G}_{\mathrm{R}} = \begin{cases}\mathbf{F}^H, & M_{\mathrm{T}} < N_{\mathrm{T}},\\ \mathbf{F}^H(\mathbf{F}\mathbf{F}^H)^{-1}, & M_{\mathrm{T}} \geq N_{\mathrm{T}}.\end{cases} \tag{9}$$

The tentatively estimated channel matrices $\mathbf{R}_{k_0}$ and $\mathbf{R}_{k_0+1}$ are then input into the SF-CNN simultaneously, which outputs the estimated channel matrices $\hat{\mathbf{H}}_{k_0}$ and $\hat{\mathbf{H}}_{k_0+1}$ through the mapping relationship

$$\left\{\hat{\mathbf{H}}_{k_0}, \hat{\mathbf{H}}_{k_0+1}\right\} = f_\Phi\left(\mathbf{R}_{k_0}, \mathbf{R}_{k_0+1}; \Phi\right), \tag{10}$$

where $\Phi$ denotes the parameter set of the SF-CNN.

*2) SF-CNN Offline Training:* For the proposed SF-CNN, the training set consisting of $N_{\mathrm{tr}}$ samples is generated according to certain channel model in the simulation environment with $(\underline{\mathbf{R}}_i, \underline{\mathbf{H}}_i)$ denoting the $i$th sample, where $\underline{\mathbf{R}}_i$ is the input data and $\underline{\mathbf{H}}_i$ is the target data. $\underline{\mathbf{R}}_i \in \mathbb{C}^{N_{\mathrm{R}}\times N_{\mathrm{T}}\times 2}$ is a three-dimensional matrix composed of $\mathbf{R}_{k_0'}^i, \mathbf{R}_{k_0'+1}^i \in \mathbb{C}^{N_{\mathrm{R}}\times N_{\mathrm{T}}}$, which are the tentatively estimated channel matrices at subcarrier $k_0'$ and $k_0' + 1$ collected through (7) with $k_0' \in \{1, 2, \ldots, K-1\}$. $\underline{\mathbf{H}}_i \in$

TABLE I
ARCHITECTURE OF THE SF-CNN

| Layer | Type | Number of filters | Size of filters | Activation function |
|---|---|---|---|---|
| 1 | Input | - | - | - |
| 2∼10 | Conv. | 64 | $3 \times 3$ | ReLU |
| 11 | Output | $2Q$ | $3 \times 3$ | tanh |

TABLE II
SF-CNN PARAMETER SETTINGS

| $l$ | $M_{1,l}$ | $M_{2,l}$ | $F_l$ | $N_{l-1}$ | $N_l$ |
|---|---|---|---|---|---|
| 1 | 16 | 32 | 3 | 4 | 64 |
| $2 \sim 9$ | 16 | 32 | 3 | 64 | 64 |
| 10 | 16 | 32 | 3 | 64 | 4 |

$\mathbb{C}^{N_\mathrm{R} \times N_\mathrm{T} \times 2}$ is also a three-dimensional matrix composed of $\frac{\mathbf{H}^i_{k'_0}}{c}, \frac{\mathbf{H}^i_{k'_0+1}}{c} \in \mathbb{C}^{N_\mathrm{R} \times N_\mathrm{T}}$, where $\mathbf{H}^i_{k'_0}$ and $\mathbf{H}^i_{k'_0+1}$ are the corresponding true channel matrices. $c > 0$ is a scaling constant to make the value range of the real and imaginary parts of all the target data, $\underline{\mathbf{H}}_i$, match the activation function applied in the output layer of the SF-CNN. Then $\underline{\mathbf{R}}_i$ is fed into the SF-CNN to approximate the corresponding scaled true channels $\underline{\mathbf{H}}_i$.

For the mmWave massive MIMO systems, we use $N_\mathrm{T} = 32$, $N_\mathrm{R} = 16$ as a typical example. As shown in Fig. 1, the SF-CNN receives the tentatively estimated complex channel matrices, $\mathbf{R}^i_{k'_0} \in \mathbb{C}^{16 \times 32}$ and $\mathbf{R}^i_{k'_0+1} \in \mathbb{C}^{16 \times 32}$, as the input and separates their real and imaginary parts so that four $16 \times 32$ real-valued matrices are obtained. In the subsequent convolutional layer, the four matrices are processed by 64 $3 \times 3 \times 4$ convolutional filters with the rectified linear unit (ReLU) activation function to generate 64 $16 \times 32$ real-valued matrices. Zero padding (ZP) is used when processing each feature matrix so that its dimension maintains unchanged after convolution. Then a batch normalization (BN) layer is added to avoid the gradient diffusion and overfitting. For the next eight convolutional layers, each uses 64 $3 \times 3 \times 64$ filters to operate ZP convolution with the feature matrices passed by the previous layer and outputs 64 $16 \times 32$ real-valued feature matrices. ReLU activation function is applied for these eight layers, each of which is followed by a BN layer. The output layer uses four $3 \times 3 \times 64$ convolutional filters to process the 64 $16 \times 32$ real-valued feature matrices and obtains the estimated real and imaginary parts of the scaled channel matrices at the $k'_0$th and $(k'_0+1)$th subcarrier, i.e., $\frac{\mathrm{Re}(\hat{\mathbf{H}}^i_{k'_0})}{c}$, $\frac{\mathrm{Im}(\hat{\mathbf{H}}^i_{k'_0})}{c}$, $\frac{\mathrm{Re}(\hat{\mathbf{H}}^i_{k'_0+1})}{c}$, and $\frac{\mathrm{Im}(\hat{\mathbf{H}}^i_{k'_0+1})}{c}$. Hyperbolic tangent activation function is used in the output layer to map the output into interval $[-1, 1]$. After scaling up and combining the corresponding real and imaginary parts, the $16 \times 32$ complex-valued estimated channel matrices, $\hat{\mathbf{H}}^i_{k'_0}$ and $\hat{\mathbf{H}}^i_{k'_0+1}$, are obtained. Table I lists the detailed architecture of the SF-CNN.

The objective of the offline training for the SF-CNN is to minimize the MSE loss function

$$\mathrm{MSE}_\mathrm{Loss} = \frac{1}{N_\mathrm{tr} c^2} \sum_{i=1}^{N_\mathrm{tr}} \sum_{q=1}^{2} \left\| \mathbf{H}^i_{k'_0+q-1} - \hat{\mathbf{H}}^i_{k'_0+q-1} \right\|_F^2. \quad (11)$$

The design of the SF-CNN architecture draws from CNN based image processing and considers the specific channel estimation task in this paper. The SF-CNN is used for channel denoising and thus we set the size of feature maps of each layer as $N_\mathrm{R} \times N_\mathrm{T}$. Nine convolutional hidden layers are used to fully uncover the inherent structure of the channel. According to [23], we adopt multiple convolutional filters with a very small size, i.e., $3 \times 3$, to achieve good channel estimation performance with the low complexity. Based on our simulation trials, further increasing the number of convolutional layers or the number of convolutional filters does not bring major performance improvement but causes much higher complexity for CNN training and testing.

*3) Online Deployment Issue:* After the offline training, the SF-CNN as well as the TE module will be deployed at the receiver to output the estimated channel matrices, $\hat{\mathbf{H}}_{k_0}, \hat{\mathbf{H}}_{k_0+1}, \ldots, \hat{\mathbf{H}}_{k_0+Q-1}$, by jointly processing the pilot matrices, $\mathbf{Y}_{k_0}, \mathbf{Y}_{k_0+1}, \ldots, \mathbf{Y}_{k_0+Q-1}$. If the actual channel model differs from what is used to generate the training set, a straightforward solution is fine-tuning. But it is difficult to collect the true channel and therefore the estimated channel is used instead. It is clear that using more power or longer pilot sequence makes the estimated channel closer to the true channel, which, however, increases the overhead for online fine-tuning. Fortunately, as shown by Fig. 6 and Fig. 7 in Section V, the offline trained SF-CNN is quite robust to most of the new channel statistics that are not observed before. This implies that further online fine-tuning only provides marginal performance improvement and hence might not be necessary.

### B. Complexity Analysis

In this subsection, we analyze the computational complexity of the proposed SF-CNN based channel estimation in testing stage and compare it with the non-ideal MMSE using estimated covariance matrix. The required number of floating point operations (FLOPs) is used as the metric.

For the proposed approach, the FLOPs come from the TE module processing in (7) and SF-CNN. By assuming $M_\mathrm{T} = N_\mathrm{T}$ and $M_\mathrm{R} = N_\mathrm{R}$, the matrix product in (7) requires FLOPs of $C_\mathrm{TE} \sim \mathcal{O}(QN_\mathrm{T}N_\mathrm{R}(N_\mathrm{T} + N_\mathrm{R}))$ [24]. According to [25], the required FLOPs of SF-CNN processing is $C_\text{SF-CNN} \sim \mathcal{O}(\sum_{l=1}^{L_\mathrm{c}} M_{1,l} M_{2,l} F_l^2 N_{l-1} N_l)$, where $L_\mathrm{c}$ is number of convolutional layers, $M_{1,l}$ and $M_{2,l}$ denote the number of rows and columns of each feature map output by the $l$th layer, $F_l$ is the side length of the filters used by the $l$th layer, $N_{l-1}$ and $N_l$ denote the numbers of input and output feature maps of the $l$th layer. Specifically, these parameters are listed in Table II based on the SF-CNN offline training mentioned above. Then the computational complexity of the proposed SF-CNN based channel estimation is given by

$$C_\text{SF-CNN-CE} \sim \mathcal{O}\left( QN_\mathrm{T}N_\mathrm{R}(N_\mathrm{T} + N_\mathrm{R}) + N_\mathrm{T}N_\mathrm{R} \sum_{l=1}^{L_\mathrm{c}} F_l^2 N_{l-1} N_l \right), \quad (12)$$

For the MMSE channel estimation, least-square (LS) channel estimation needs to be first performed causing FLOPs of $C_{\text{LS}} \sim \mathcal{O}(QN_{\text{T}}^2N_{\text{R}}^2)$. The channel covariance matrix is then calculated based on the LS channel estimation once per channel realization, which requires the computational complexity of $C_{\text{MMSE},1} \sim \mathcal{O}(Q^2N_{\text{T}}^2N_{\text{R}}^2)$ if considering both spatial and frequency channel statistics. Finally, the LS channel estimation is refined by the covariance matrix and the corresponding FLOPs is $C_{\text{MMSE},2} \sim \mathcal{O}(Q^3N_{\text{T}}^3N_{\text{R}}^3)$. Therefore, the overall computational complexity of MMSE is

$$C_{\text{MMSE}} \sim \mathcal{O}(Q^3N_{\text{T}}^3N_{\text{R}}^3). \tag{13}$$

It is hard to compare $C_{\text{SF-CNN-CE}}$ with $C_{\text{MMSE}}$ straightforwardly in general since the former depends on $L_{\text{c}}$, $F_l$, $N_{l-1}$, and $N_l$ besides $Q$, $N_{\text{T}}$ and $N_{\text{R}}$. If $N_{\text{T}} = 32$, $N_{\text{R}} = 16$, $Q = 2$ and other parameters for the SF-CNN are listed in Table II, the computational complexity of the proposed SF-CNN based approach is in the order of magnitude of $10^8$ while MMSE needs a higher complexity in the order of magnitude of $10^9$. In addition, the SF-CNN is able to run in a more efficient parallel manner and the runtime of a channel realization is only $1.47 \times 10^{-4}$ seconds by using NVIDIA GeForce GTX 1080 Ti GPU. By comparison, the MMSE consumes the time of about $6.14 \times 10^{-2}$ seconds per channel realization on the Intel(R) Core(TM) i7-3770 CPU.

## IV. SFT-CNN and SPR-CNN-Based Channel Estimation

In this section, we first develop a SFT-CNN based channel estimation approach, which further incorporates channel temporal correlation into the SF-CNN. Then the SFT-CNN is modified to the SPR-CNN to mitigate the huge spatial pilot overhead caused by large-scale antenna arrays.

For time-varying channels, the frequency-domain channel at the $k$th subcarrier in (4) becomes [7]

$$\mathbf{H}_k(t) = \sqrt{\frac{N_{\text{T}}N_{\text{R}}}{L}} \sum_{l=1}^{L} \alpha_l e^{-j2\pi(\tau_l f_s \frac{k}{K} - \nu_l t)} \mathbf{a}_{\text{R}}(\varphi_l)\mathbf{a}_{\text{T}}^H(\phi_l), \tag{14}$$

where $\nu_l$ denotes the Doppler shift of the $l$th path.

According to [26], [27], the temporal correlation between channels in successive coherence intervals can be modeled as Gauss-Markov distribution

$$\mathbf{H}_k[n] = \rho\mathbf{H}_k[n-1] + \sqrt{1-\rho^2}\mathbf{\Theta}[n], \quad n \in \mathbb{N}_+ \tag{15}$$

where $\mathbf{H}_k[n] = \mathbf{H}_k(nT)$ is the discrete-time version of $\mathbf{H}_k(t)$ with $T$ denoting the length of the coherence interval, $0 \le \rho \le 1$ denotes the temporal correlation coefficient, and $\mathbf{\Theta}[n]$ is a random matrix accounting for the innovation process with unit variance for each entry. (15) clearly demonstrates that some inherence underlays the channel variation from the previous coherence interval to the current one and this correlation can be also exploited to improve the channel estimation accuracy in addition to the spatial and frequency correlation. In the following, we first elaborate the SFT-CNN based channel estimation.

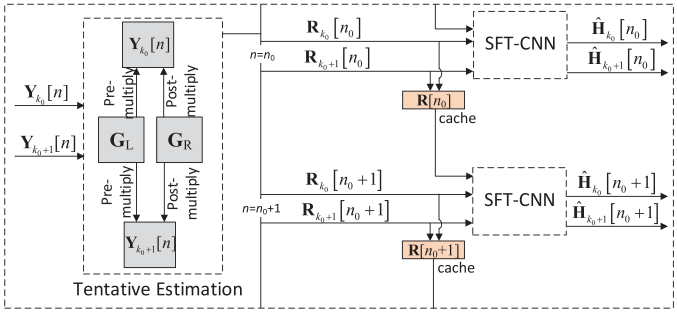



Fig. 2. SFT-CNN based channel estimation.

### A. SFT-CNN based Channel Estimation

As shown in Fig. 2, we still consider channel estimation at $Q$ $(= 2)$ adjacent subcarriers, $k_0$ and $k_0 + 1$, for ease of illustration. In time-varying channels, the received pilots after combining at the receiver in (6) becomes

$$\mathbf{Y}_k[n] = \sqrt{P}\mathbf{W}^H\mathbf{H}_k[n]\mathbf{F} + \tilde{\mathbf{N}}_k[n], \quad k \in \{k_0, k_0+1\}. \tag{16}$$

Similar to SF-CNN based channel estimation, $\mathbf{Y}_k[n]$ is then processed by the TE module, which generates the tentatively estimated channel matrices sequentially as

$$\mathbf{R}_k[n] = \sqrt{P}\mathbf{G}_{\text{L}}\mathbf{W}^H\mathbf{H}_k[n]\mathbf{F}\mathbf{G}_{\text{R}} + \mathbf{G}_{\text{L}}\tilde{\mathbf{N}}_k[n]\mathbf{G}_{\text{R}}. \tag{17}$$

Then a SFT-CNN further refines these tentatively estimated channel matrices by exploiting the spatial, frequency, and temporal correlation of channels simultaneously. As shown in Fig. 2, we capture $S$ $(= 2)$ successive coherence intervals, $n_0$ and $(n_0 + 1)$, to describe the channel estimation procedure. In the $n_0$th coherence interval, the tentatively estimated channel matrices, $\mathbf{R}_{k_0}[n_0]$ and $\mathbf{R}_{k_0+1}[n_0]$, are input into the SFT-CNN. A copy of $\mathbf{R}_{k_0}[n_0]$ and $\mathbf{R}_{k_0+1}[n_0]$ is stored in the cache in order to be used in the next coherence interval. In $(n_0 + 1)$th coherence interval, the SFT-CNN receives tentatively estimated channel matrices, $\mathbf{R}_{k_0}[n_0 + 1]$ and $\mathbf{R}_{k_0+1}[n_0 + 1]$, as well as fetches $\mathbf{R}_{k_0}[n_0]$ and $\mathbf{R}_{k_0+1}[n_0]$ from the cache to perform joint processing and obtain the estimated channel matrices as

$$\left\{\hat{\mathbf{H}}_{k_0}[n_0+1], \hat{\mathbf{H}}_{k_0+1}[n_0+1]\right\} = f_\Psi(\mathbf{R}_{k_0}[n_0], \mathbf{R}_{k_0+1}[n_0], \mathbf{R}_{k_0}[n_0{+}1], \mathbf{R}_{k_0+1}[n_0{+}1]; \Psi), \tag{18}$$

where $\Psi$ denotes the parameter set of the SFT-CNN. Meanwhile, the cache is updated by replacing $\mathbf{R}_{k_0}[n_0]$ and $\mathbf{R}_{k_0+1}[n_0]$ with $\mathbf{R}_{k_0}[n_0 + 1]$ and $\mathbf{R}_{k_0+1}[n_0 + 1]$. In each coherence interval, the same SFT-CNN is used since it has learned the general channel temporal correlation instead of the specific relationship between channels in two successive coherence intervals.

After summarizing the channel estimation procedure, we focus on the offline training of the SFT-CNN. Similar to SF-CNN, the training set consisting of $N_{\text{tr}}$ samples is generated according to certain channel model in the simulation environment with $(\underline{\mathbf{R}}_i, \underline{\mathbf{H}}_i)$ denoting the $i$th sample. $\underline{\mathbf{R}}_i \in \mathbb{C}^{N_{\text{R}} \times N_{\text{T}} \times 4}$ is a three-dimensional matrix composed of the tentatively estimated channel matrices in the $n_0'$th and $(n_0' + 1)$th

TABLE III
ARCHITECTURE OF THE LSTM-CNN

| Layer | Type | Number of filters | Size of filters | Activation function | Recurrent activation function |
|---|---|---|---|---|---|
| 1 | Input | - | - | - | - |
| 2∼4 | Conv. LSTM | 4 | $3 \times 3$ | tanh | tanh |
| 5∼11 | Conv. | 64 | $3 \times 3$ | ReLU | - |
| 12 | Output | $2Q$ | $3 \times 3$ | tanh | - |

coherence intervals collected through (17), that is $\underline{\mathbf{R}}_i = [\mathbf{R}^i_{k'_0}[n'_0], \mathbf{R}^i_{k'_0+1}[n'_0], \mathbf{R}^i_{k'_0}[n'_0+1], \mathbf{R}^i_{k'_0+1}[n'_0+1]]$ with $n'_0 \in \mathbb{N}_+$. $\underline{\mathbf{H}}_i \in \mathbb{C}^{N_\mathrm{R} \times N_\mathrm{T} \times 2}$ is also a three-dimensional matrix composed of the scaled true channel matrices in the $(n'_0+1)$th coherence interval, that is $\underline{\mathbf{H}}_i = \left[\frac{\mathbf{H}^i_{k'_0}[n'_0+1]}{c}, \frac{\mathbf{H}^i_{k'_0+1}[n'_0+1]}{c}\right]$. As before, $c > 0$ is the scaling constant to make the value range of the real and imaginary parts of all the target data, $\underline{\mathbf{H}}_i$, match the activation function applied in the output layer of the SFT-CNN. Then $\underline{\mathbf{R}}_i$ is fed into the SFT-CNN to approximate the corresponding scaled true channels $\underline{\mathbf{H}}_i$. The architecture of the SFT-CNN is similar to the SF-CNN except that it has the additional input from the previous coherence interval. With the estimated scaled channel matrices, $\frac{\hat{\mathbf{H}}^i_{k'_0}[n'_0+1]}{c}, \frac{\hat{\mathbf{H}}^i_{k'_0+1}[n'_0+1]}{c}$, the objective of the SFT-CNN offline training is to minimize the MSE loss function

$$\mathrm{MSE}_{\mathrm{Loss}} = \frac{1}{N_{\mathrm{tr}} c^2} \sum_{i=1}^{N_{\mathrm{tr}}} \sum_{q=1}^{2} \left\| \mathbf{H}^i_{k'_0+q-1}[n'_0+1] - \hat{\mathbf{H}}^i_{k'_0+q-1}[n'_0+1] \right\|_F^2. \tag{19}$$

Compared to SF-CNN, SFT-CNN only increases the computational complexity for the first convolutional layer by $S$ times, which is a quite minor part in the total computational complexity according to (12) and Table II. In contrast, if further incorporating temporal correlation, the complexity of MMSE channel estimation in (13) becomes

$$C_{\mathrm{MMSE}} \sim \mathcal{O}(S^3 Q^3 N_\mathrm{T}^3 N_\mathrm{R}^3), \tag{20}$$

which will be increased significantly even with $S = 2$. Therefore, SFT-CNN provides a simple and efficient way to utilize the channel spatial, frequency, and temporal correlation simultaneously to improve the channel estimation accuracy.

### B. SFT-CNN or LSTM-CNN?

The LSTM-CNN would naturally come to mind when we consider temporal correlation. In this subsection, we will compare SFT-CNN and LSTM-CNN in terms of estimation accuracy and complexity, which explains why the SFT-CNN in used instead of LSTM-CNN.

*1) Architecture:* The LSTM-CNN consists of one input layer, three convolutional LSTM layers, seven convolutional layers, and one output layer, where the processing of each convolutional LSTM layer follows the operation in [28]. The detailed architecture of the LSTM-CNN is listed in Table III.

*2) Channel Estimation Accuracy:* To measure the channel estimation performance, we use the normalized MSE (NMSE),

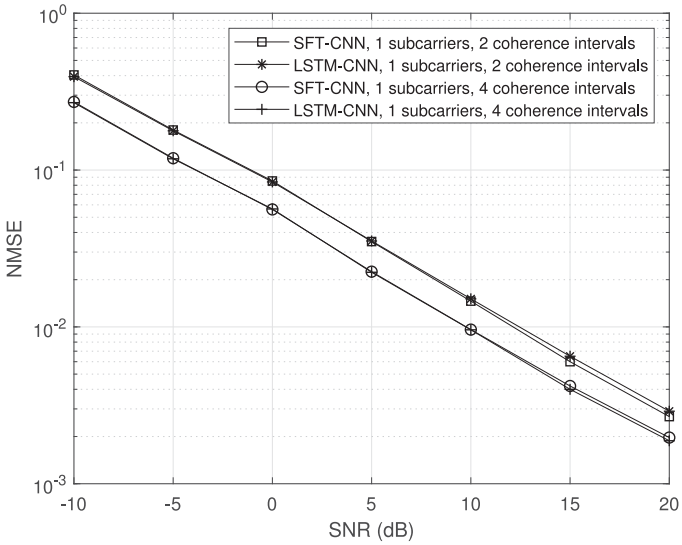



Fig. 3. NMSE for SFT-CNN and LSTM-CNN based channel estimation.

TABLE IV
TIME COMPLEXITY OF TRAINING AND TESTING

| | Training time seconds/epoch | | Testing time seconds/channel realization | |
|---|---|---|---|---|
| | $N_c = 2$ | $N_c = 4$ | $N_c = 2$ | $N_c = 4$ |
| SFT-CNN | 22 | 23 | $1.51 \times 10^{-4}$ | $1.7 \times 10^{-4}$ |
| LSTM-CNN | 34 | 48 | $3.53 \times 10^{-4}$ | $5.14 \times 10^{-4}$ |

defined as,

$$\mathrm{NMSE} = \mathbb{E}_{\mathbf{H}} \left\{ \|\mathbf{H} - \hat{\mathbf{H}}\|_F^2 / \|\mathbf{H}\|_F^2 \right\}, \tag{21}$$

where $\mathbf{H}$ and $\hat{\mathbf{H}}$ refer to the true and estimated channels, respectively. Fig. 3 plots the NMSE performance of SFT-CNN and LSTM-CNN based channel estimation, where two and four coherence intervals are involved, respectively. From this figure, SFT-CNN and LSTM-CNN achieve almost the same performance in the whole signal-to-noise ratio (SNR) regime.

*3) Training and Testing Complexity:* With the same NMSE performance, the complexity becomes an important metric for method selection. Table IV provides the time complexity of training and testing for SFT-CNN and LSTM-CNN on the NVIDIA GeForce GTX 1080 Ti GPU, where $N_c$ denotes the number of coherence intervals involved in the CNNs. It is clear that LSTM-CNN consumes much more time than SFT-CNN in both training and testing stages.

Based on the above discussion, the LSTM-CNN has no performance advantages in exploiting channel temporal correlation but requires much higher complexity for training and testing. Since the SFT-CNN can also capture the channel temporal correlation efficiently with relatively low complexity, this simple pure CNN architecture is preferable for the channel estimation task in this paper.

### C. SPR-CNN-Based Channel Estimation

Large-scale array antennas at both the transmitter and the receiver incur huge pilot overhead in spatial domain. In this

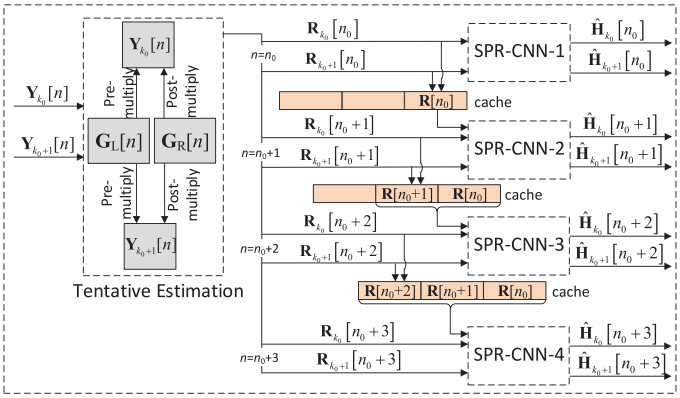



Fig. 4. SPR-CNN-based channel estimation.

subsection, we design the SPR-CNN based channel estimation, which uses much fewer pilots but still guarantees the fairly good accuracy.

The basic idea of the SPR-CNN based channel estimation can be summarized as follows:

1) Group $D$ successive coherence intervals as a CEU, in which channel correlation is utilized to reduce the spatial pilot overhead. Different CEUs are non-overlapped.
2) In the first coherence interval of each CEU, use full spatial pilot overhead for channel estimation.[2] Then the pilot overhead is reduced in the subsequent coherence intervals.
3) For the first coherence interval, the receiver uses the currently received pilots to estimate the current channels. For the rest of coherence intervals, the receiver uses the currently and all previously received pilots in this CEU to jointly estimate the current channels.

Here is the detailed channel estimation procedure. Different beamforming and combining matrices are employed in different coherence intervals of each CEU. As shown in Fig. 4, the received pilots after combining at the receiver in (16) becomes

$$\mathbf{Y}_k[n] = \sqrt{P}\mathbf{W}^H[n]\mathbf{H}_k[n]\mathbf{F}[n] + \tilde{\mathbf{N}}_k[n], \tag{22}$$

where $\mathbf{F}[n] \in \mathbb{C}^{N_\mathrm{T}\times M_\mathrm{T}[n]}$ and $\mathbf{W}[n] \in \mathbb{C}^{N_\mathrm{R}\times M_\mathrm{R}[n]}$ denote the beamforming matrix and combining matrix, respectively, in the $n$th coherence interval. The corresponding spatial pilot overhead is given by

$$p[n] = M_\mathrm{T}[n]\left\lceil \frac{M_\mathrm{R}[n]}{N_\mathrm{R}^\mathrm{RF}} \right\rceil. \tag{23}$$

From (23), the spatial pilot overhead can be saved by reducing $M_\mathrm{T}[n]$ or/and $M_\mathrm{R}[n]$.

$\mathbf{Y}_k[n]$ is first processed by the TE module and the tentatively estimated channel matrix is given by

$$\begin{aligned}\mathbf{R}_k[n] &= \mathbf{G}_\mathrm{L}[n]\mathbf{Y}_k[n]\mathbf{G}_\mathrm{R}[n] \\ &= \sqrt{P}\mathbf{G}_\mathrm{L}[n]\mathbf{W}^H[n]\mathbf{H}_k[n]\mathbf{F}[n]\mathbf{G}_\mathrm{R}[n] \\ &\quad + \mathbf{G}_\mathrm{L}[n]\tilde{\mathbf{N}}_k[n]\mathbf{G}_\mathrm{R}[n],\end{aligned} \tag{24}$$

where $\mathbf{G}_\mathrm{L}[n]$ and $\mathbf{G}_\mathrm{R}[n]$ are also changed along with the coherence interval index and are expressed as

$$\mathbf{G}_\mathrm{L}[n] = \begin{cases}\mathbf{W}[n], & M_\mathrm{R}[n] < N_\mathrm{R}, \\ (\mathbf{W}[n]\mathbf{W}^H[n])^{-1}\mathbf{W}[n], & M_\mathrm{R}[n] \geq N_\mathrm{R},\end{cases} \tag{25}$$

and

$$\mathbf{G}_\mathrm{R}[n] = \begin{cases}\mathbf{F}^H[n], & M_\mathrm{T}[n] < N_\mathrm{T}, \\ \mathbf{F}^H[n](\mathbf{F}[n]\mathbf{F}^H[n])^{-1}, & M_\mathrm{T}[n] \geq N_\mathrm{T}.\end{cases} \tag{26}$$

Then the $\mathbf{R}_k[n]$ is processed by the CNN estimation part. We consider that every $D\ (=4)$ successive coherence intervals are grouped as a CEU and capture a certain CEU with the $n_0$th to the $(n_0+3)$th coherence intervals, as shown in Fig. 4. There are four SFT-CNNs with different input and output, called SPR-CNN-1, 2, 3, and 4, respectively. In the $n_0$th coherence interval, full spatial pilot overhead, i.e., $M_\mathrm{T}[n] = N_\mathrm{T}$ and $M_\mathrm{R}[n] = N_\mathrm{R}$, is used to provide accurate channel information for all coherence intervals of this CEU. After the TE module, $\mathbf{R}_{k_0}[n_0]$ and $\mathbf{R}_{k_0+1}[n_0]$ are input into the SPR-CNN-1, which generates the finally estimated channel matrices, $\hat{\mathbf{H}}_{k_0}[n_0]$ and $\hat{\mathbf{H}}_{k_0+1}[n_0]$. Meanwhile, a copy of $\mathbf{R}_{k_0}[n_0]$ and $\mathbf{R}_{k_0+1}[n_0]$ is stored in the cache to provide additional channel information for the channel estimation of subsequent coherence intervals. In the $(n_0+1)$th coherence interval, the dimensions of $\mathbf{F}[n]$ and $\mathbf{W}[n]$ will be reduced to save the pilot overhead, i.e., $M_\mathrm{T}[n] < N_\mathrm{T}$ and $M_\mathrm{R}[n] < N_\mathrm{R}$.[3] $\mathbf{R}_{k_0}[n_0]$ and $\mathbf{R}_{k_0+1}[n_0]$ stored in the cache along with $\mathbf{R}_{k_0}[n_0+1]$ and $\mathbf{R}_{k_0+1}[n_0+1]$ are simultaneously input into SPR-CNN-2 to obtain $\hat{\mathbf{H}}_{k_0}[n_0+1]$ and $\hat{\mathbf{H}}_{k_0+1}[n_0+1]$. A copy of $\mathbf{R}_{k_0}[n_0+1]$ and $\mathbf{R}_{k_0+1}[n_0+1]$ is also stored in the cache in addition to $\mathbf{R}_{k_0}[n_0]$ and $\mathbf{R}_{k_0+1}[n_0]$. All matrices stored in the cache are used for the joint channel estimation of the $(n_0+2)$th coherence interval. Channel estimations in the $(n_0+2)$th and $(n_0+3)$th coherence intervals are similar to that in the $(n_0+1)$th coherence interval. After channel estimation in the $(n_0+3)$th coherence interval, the cache will be emptied and then used for the next CEU. From Fig. 4, four different SPR-CNNs are employed for respective coherence intervals in a CEU and reused for all CEUs. The architecture and training process of the SPR-CNNs are similar to SFT-CNN except that the numbers of input matrices are different for different SPR-CNNs. An intuitional description of the SPR-CNN based channel estimation is given by Algorithm 1.

Among the four SPR-CNNs, SPR-CNN-4 has the highest complexity with the most input matrices. But it just increases the complexity of the first convolutional layer by $D\ (=4)$ times compared to the SF-CNN in Section III, which causes limited impact on the total computational complexity. Therefore, SPR-CNN based channel estimation saves the spatial pilot overhead effectively while only increases the complexity moderately.

We design SPR-CNN based channel estimation aiming to reduce the pilot overhead in spatial domain significantly while still guarantee fairly good channel estimation accuracy. In each

[2]Full spatial pilot overhead means that the number of beamforming vectors is equal to the number of transmit antennas and the number of combining vectors is equal to the number of receive antennas.

[3]Different pilot overheads may be used in different coherence intervals of a CEU. But for each coherence interval, it and its counterparts in other CEUs should use the same pilot overhead.

**Algorithm 1:** SPR-CNN based Channel Estimation.

**Input:** The total number of CEUs $M_{\text{CEU}}$, the number of coherence intervals in each CEU $D$, spatial pilot overhead from the second to $D$th coherence intervals of each CEU

**Output:** Estimated channel matrices

**Procedure:**

Initialize the CEU and coherence interval indices as $m = 1$ and $d = 1$;
Train $D$ different SPR-CNNs for the first to $D$th coherence intervals;
**for** $m \in [1, M_{\text{CEU}}]$ **do**
  **for** $d \in [1, D]$ **do**
    **if** $d = 1$ **then**
      Use full spatial pilot overhead;
      Tentatively estimate channels according to (24);
      Store the tentatively estimated channel matrices in the cache;
      Input the tentatively estimated channel matrices of the first coherence interval into SPR-CNN-1 to obtain the estimated channel matrices of the first coherence interval;
    **else**
      Use reduced spatial pilot overhead;
      Tentatively estimate channels according to (24);
      Store the tentatively estimated channel matrices in the cache (Invalid for $d = D$);
      Input the tentatively estimated channel matrices from the first to $d$th coherence intervals into SPR-CNN-$d$ to obtain the estimated channel matrices of the $d$th coherence interval;
    **end if**
  **end for**
  Empty the cache and reset $d = 1$
**end for**
**return** the estimated channel matrices of $M_{\text{CEU}}$ CEUs

CEU with $D$ coherence intervals, full pilot overhead is used in the first coherence interval while reduced pilot overhead is used in the remaining coherence intervals. The first coherence interval provides complete channel information that is quite helpful to channel estimation of all coherence intervals in this CEU. It is clear that the average pilot overhead of a CEU reduces as $D$ increases. However, increasing $D$ will also degrade the average channel estimation accuracy since the effect of the complete channel information provided by the first coherence interval weakens along with the evanescent temporal correlation. Therefore, our architecture of SPR-CNN includes four coherence intervals in a CEU to achieve a good tradeoff between the pilot overhead and the estimation accuracy. In addition, if using the LSTM-CNN to perform the channel estimation in Fig. 4, four LSTM-CNNs with the architecture listed in Table III are needed. As discussed in Section IV-B, the LSTM-CNN has no performance superiority over the simple SFT-CNN architecture but causes much higher complexity and thus is not suitable for the SPR-CNN based channel estimation.

## V. Numerical Results

In this section, we present simulation results of the proposed CNN based channel estimation approaches and compare them with non-ideal MMSE using the estimated covariance matrix and ideal MMSE using the true covariance matrix. We set the number of antennas at the transmitter, $N_{\text{T}} = 32$, the number of antennas at the receiver, $N_{\text{R}} = 16$, and the numbers of RF chains at the transmitter and the receiver $N_{\text{T}}^{\text{RF}} = N_{\text{R}}^{\text{RF}} = 2$. $\mathbf{F}$(or $\mathbf{F}[n]$) and $\mathbf{W}$(or $\mathbf{W}[n]$) are set as the first $M_{\text{T}}$ (or $M_{\text{T}}[n]$) columns of an $N_{\text{T}} \times N_{\text{T}}$ discrete Fourier transform (DFT) matrix and the first $M_{\text{R}}$ (or $M_{\text{R}}[n]$) columns of an $N_{\text{R}} \times N_{\text{R}}$ DFT matrix. In Section V-A and Section V-B, we set $M_{\text{T}} = 32$ and $M_{\text{R}} = 16$. The settings of $M_{\text{T}}[n]$ and $M_{\text{R}}[n]$ will be introduced in Section V-C.

The channel data are generated according to the 3rd Generation Partnership Project (3GPP) TR 38.901 Release 15 channel model [29]. Specifically, we use the clustered delay line models with the carrier frequency, $f_c = 28$ GHz, the sampling rate, $f_s = 100$ MHz, the number of main paths, $L = 3$, and the number of subcarriers, $K = 64$.

For the SF-CNN, the training set, validation set, and testing set contain 81,000, 9,000, and 19,000 samples, respectively. The parameters of each layer are set as Table I. Adam is used as the optimizer. The epochs are set as 800 while the corresponding learning rates are $10^{-4}$ for the first 200 epochs, $5 \times 10^{-5}$ for the next 400 epochs, and $10^{-5}$ for the last 200 epochs, respectively.[4] The batch size is 128. The scaling constant is set as $c = 2$. The SFT-CNN and SPR-CNN use the same parameters as the SF-CNN except that the numbers of input matrices are different.

### A. SF-CNN-Based Channel Estimation

Figure 5 shows the NMSE performance versus SNR of the proposed SF-CNN based channel estimation and MMSE channel estimation over two adjacent subcarriers in the urban micro (UMi) street non-line of sight (NLOS) scenario.[5] The performance of the SF-CNN based approach at single subcarrier is also plotted to demonstrate that frequency correlation is helpful to improve the channel estimation accuracy. Through offline training, the SF-CNN based channel estimation outperforms the non-ideal MMSE with estimated covariance matrix significantly yet requiring lower estimation complexity according to this figure and Section III-B. Moreover, the performance of the SF-CNN based approach is very close to the ideal MMSE with true covariance matrix, especially at the low and medium SNRs.

The robustness of the MMSE and proposed SF-CNN based approaches is shown in Fig. 6. The joint channel estimation over two subcarriers is considered. The SF-CNN is trained in

[4]Although optimizer Adam can adaptively adjust the learning rate for each epoch by considering the momentum as well as the gradients, we still manually set the learning rate at the specific epochs to speed up the convergence of offline training.

[5]We train different neural networks separately for different SNRs based on the same channel dataset. Specifically, we first generate a channel dataset according to the 3GPP TR 38.901 Release 15 channel model. Then we use these channel data to generate the received pilot signals in different SNRs, which are used to train the neural networks for corresponding SNRs. The testing set is generated similarly.

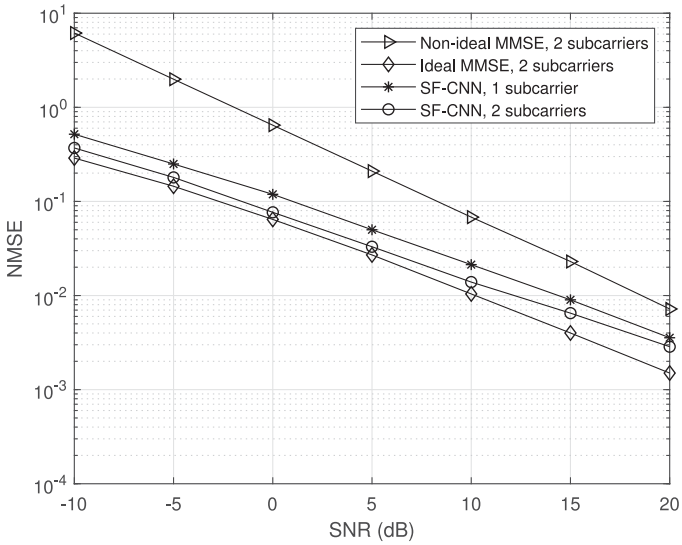



Fig. 5. NMSE versus SNR for the SF-CNN based channel estimation and MMSE channel estimation.

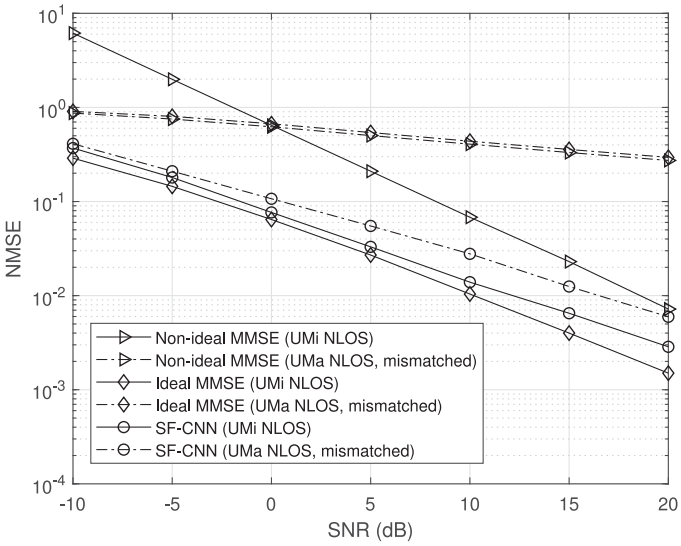



Fig. 6. Robustness of MMSE and SF-CNN based approaches to different scenarios.

the UMi street NLOS scenario and is tested in both UMi street NLOS scenario and urban macro (UMa) NLOS scenario. For the MMSE, its covariance matrix is calculated in the UMi street NLOS scenario and then the channel matrix is estimated in both UMi street NLOS scenario and UMa NLOS scenario. According to [29], the power, delay, and angle profiles of UMa NLOS scenario are quite different from those of UMi street NLOS scenario. The channel statistics are unknown to both SF-CNN and MMSE when they predict the channels in the UMa NLOS scenario. From this figure, the SF-CNN based channel estimation exhibits good robustness when facing the significantly different channel statistics. Even under the mismatched UMa NLOS scenario, the SF-CNN based approach still outperforms the non-ideal MMSE without mismatch. In contrast, due to the high dependence on channel statistics, both the ideal and non-ideal MMSE fail to cope with the change of channel statistics and suffer significant performance loss.

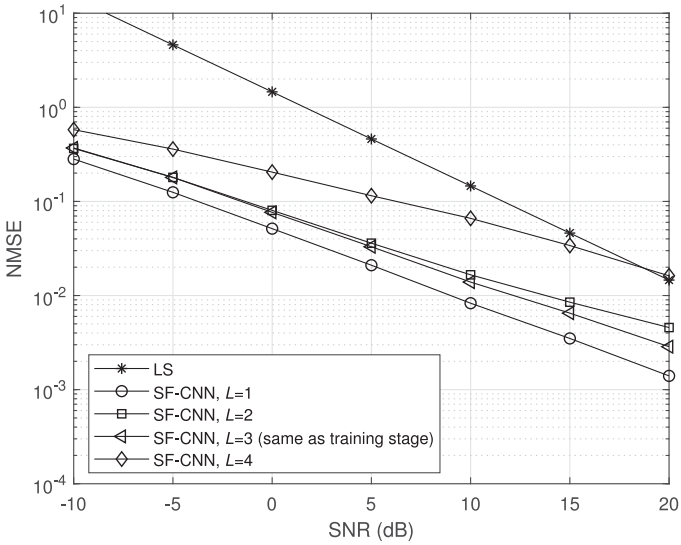



Fig. 7. Robustness of SF-CNN based approach to different numbers of main paths.

The number of main paths, $L$, may change in the real scenario. Here, we test the robustness of the SF-CNN based approach to different numbers of main paths, which is plotted in Fig. 7 with $L = 3$ in the training stage and $L = 1, 2, 3, 4$ in the testing stage. The case of $L = 3$ in the testing stage is matched with the training stage and thus acts as the baseline for comparison. LS channel estimation is also plotted as an upper bound. The SF-CNN based approach with $L = 1$ performs better than the baseline, which is because the channel structure becomes quite simple and thus facilitates the CNN denoising. When $L$ is increased to 2, the performance is slightly degraded compared to the baseline. Although there is a significant performance degeneration when $L$ becomes 4, the proposed approach still outperforms LS channel estimation when SNR $<$ 18 dB. In fact, $L = 1$ and 2 can be regarded as two special cases of $L = 3$. The offline trained SF-CNN with $L = 3$ is able to identify the channels with $L = 1$ and 2 and thus exhibits good robustness. In contrast, the case of $L = 4$ leads to a significant change of channel structure, to which the offline trained model is less robust.

### *B. SFT-CNN-Based Channel Estimation*

Figure 8 shows the NMSE performance versus SNR of the SF-CNN, SFT-CNN, and MMSE based channel estimation approaches in the UMi street NLOS scenario. The MMSE and proposed SFT-CNN based approaches utilize the channel information of the previous coherence interval to jointly estimate the current channels over two subcarriers while the SF-CNN based approach does not incorporate temporal correlation. By comparing the circle and square curves, we can clearly see the effect of temporal correlation on improving the NMSE performance with the SFT-CNN based channel estimation. The comparison between the star and square curves demonstrates that just utilizing the additional channel information of adjacent subcarrier

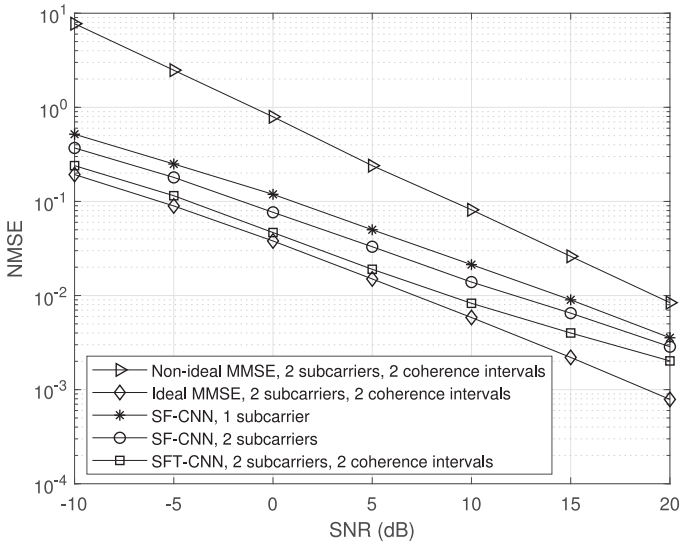



Fig. 8. NMSE versus SNR for the SFT-CNN based channel estimation and MMSE channel estimation.

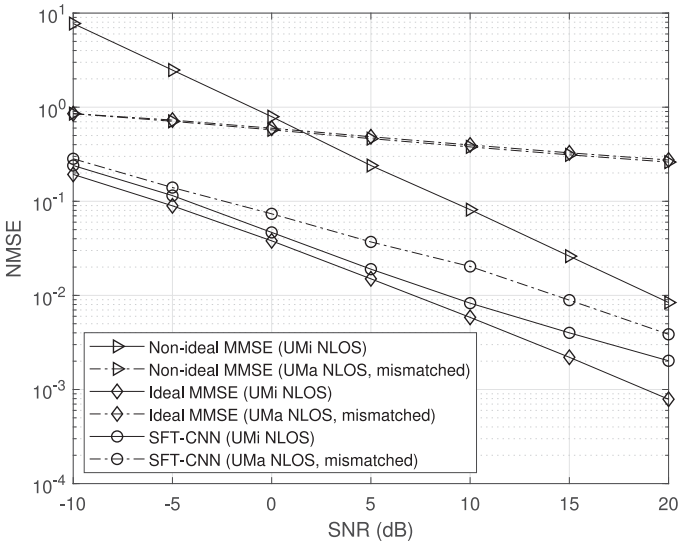



Fig. 9. Robustness of MMSE and SFT-CNN based approaches to different scenarios.

and coherence interval can improve the estimation accuracy remarkably while only increases the complexity slightly. With the same channel correlation information, the SFT-CNN based approach outperforms the non-ideal MMSE significantly in both performance and complexity at the cost of offline training. It is even very close to the ideal MMSE at the low and medium SNRs.

In Fig. 9, we demonstrate the robustness of the MMSE and proposed SFT-CNN based approaches using the channel correlation of two subcarriers and two coherence intervals. Similar to Fig. 6, both the covariance matrix calculation of the MMSE and the offline training of the SFT-CNN are performed in the UMi street NLOS scenario with the maximum Doppler spread $f_d = 1{,}400$ Hz. When the testing scenario is unchanged, all these approaches can achieve good performance. If we test them in a mismatched scenario, i.e., UMa NLOS scenario with $f_d = 1{,}800$ Hz in Fig. 9, the performance of both ideal and non-ideal MMSE severely degrades since the mismatched covariance matrix is counterproductive when refining the LS estimated channels. In contrast, the SFT-CNN has learned the more inherent channel structure and thus exhibits superior robustness to the significantly different scenarios.

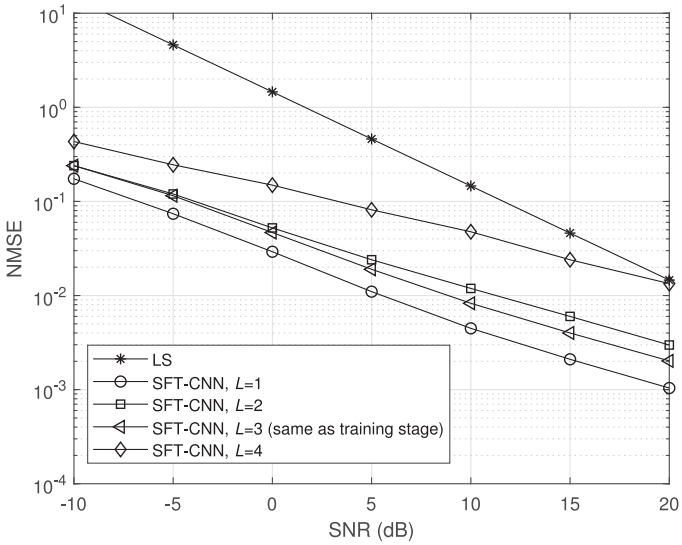



Fig. 10. Robustness of SFT-CNN based approach to different numbers of main paths.

The robustness of SFT-CNN based approach to different numbers of main paths is evaluated in Fig. 10, which exhibits similar results to Fig. 7. Nevertheless, temporal correlation is still helpful to improve the channel estimation accuracy even with mismatched number of main paths by comparing Fig. 7 and Fig. 10.[6]

### C. SPR-CNN-Based Channel Estimation

In Fig. 11, we verify the effectiveness of the SPR-CNN based channel estimation in the UMi street NLOS scenario, where every $D$ $(= 4)$ successive coherence intervals are grouped as a CEU. In each CEU, we set $M_{\mathrm{T}}[n] = N_{\mathrm{T}} = 32$, $M_{\mathrm{R}}[n] = N_{\mathrm{R}} = 16$, for $d = 1$, and $M_{\mathrm{T}}[n] = 16$, $M_{\mathrm{R}}[n] = 4$, for $d = 2, 3, 4$. So the average spatial pilot overhead of the SPR-CNN based approach is $p_{\text{SPR-CNN}} = \frac{\sum_{d=1}^{4} p[d]}{4} = 88$ and the ratio of it over the full pilot overhead is $r = \frac{p_{\text{SPR-CNN}}}{p_{\text{full}}} = \frac{88}{256} \approx \frac{1}{3}$. For fair comparison, the ideal and non-ideal MMSE based approaches in Fig. 11 also use the above parameter settings. The SF-CNN and SFT-CNN based approaches with full pilot overhead are also plotted for comparison. From the figure, the SPR-CNN based

[6]Robustness is an important issue since a robust DNN can avoid the complex online fine-tuning process. There are infinite channel statistics or scenarios in reality and a DNN robust to as many cases as possible is expected to design. This is a challenging task and needs more research attention since the settings of the NN architecture and various hyper-parameters are still unclear for the complicated mmWave MIMO channels.

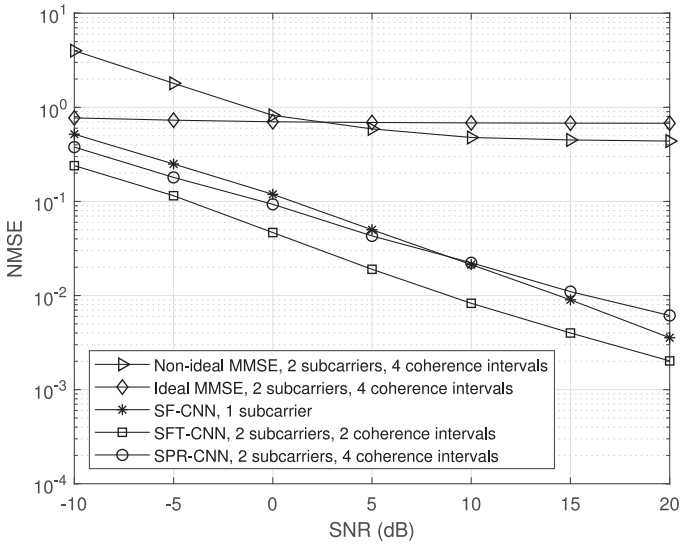



Fig. 11. NMSE versus SNR for the SPR-CNN based channel estimation and MMSE channel estimation.

channel estimation achieves comparable performance to the SF-CNN and SFT-CNN based approaches, especially at the low and medium SNR, but only requires about one third of pilot overhead at the cost of complexity. This means that the additional frequency and temporal correlation has been efficiently utilized by the SPR-CNN based approach to reduce the spatial pilot overhead significantly. On the contrary, both the ideal and non-ideal MMSE perform poorly even if using the same channel correlation information as the SPR-CNN based approach, which reveals that the proposed approach is able to tolerate the reduction of spatial pilot overhead.

## VI. Conclusion

In this paper, we have developed the deep CNN based channel estimation approaches for mmWave massive MIMO-OFDM systems. The SF-CNN based channel estimation is first proposed to simultaneously utilize spatial and frequency correlation. To further incorporate the temporal correlation in the real scenario, we develop the SFT-CNN based approach. Finally, considering the huge spatial pilot overhead caused by massive antennas, we design the SPR-CNN based channel estimation to mitigate this problem. Numerical results show that the proposed SF-CNN and SFT-CNN based approaches outperform the non-ideal MMSE estimator yet requiring lower complexity and achieve the performance very close to the ideal MMSE estimator. Even if the channel statistics are different, the proposed approaches can still achieve fairly good performance. The SPR-CNN based channel estimation is efficient to save the spatial pilot overhead significantly with minor performance loss.

The proposed CNN based channel estimation approaches can be further improved in terms of the robustness to as many channel statistics as possible. To achieve this aim, the complicated mmWave MIMO channel structure needs to be deeply dissected to provide insights for the design of DNN architecture and hyperparameters tuning.

## Acknowledgment

The authors would like to sincerely thank the support and comments from Intel Corporation and the comments from the reviewers, which have helped us improve the quality of the paper significantly.

## References

[1] P. Dong, H. Zhang, G. Y. Li, N. NaderiAlizadeh, and I. Gaspar, "Deep CNN for wideband mmWave massive MIMO channel estimation using frequency correlation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Brighton, U.K., May 2019, pp. 4529–4533.

[2] A. L. Swindlehurst, E. Ayanoglu, P. Heydari, and F. Capolino, "Millimeter-wave massive MIMO: The next wireless revolution?" *IEEE Commun. Mag.*, vol. 52, no. 9, pp. 56–62, Sep. 2014.

[3] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 186–195, Feb. 2014.

[4] H. Q. Ngo, E. G. Larsson, and T. L. Marzetta, "Energy and spectral efficiency of very large multiuser MIMO systems," *IEEE Trans. Commun.*, vol. 61, no. 4, pp. 1436–1449, Apr. 2013.

[5] B. Wang, F. Gao, S. Jin, H. Lin, and G. Y. Li, "Spatial- and frequency-wideband effects in millimeter-wave massive MIMO systems," *IEEE Trans. Signal Process.*, vol. 66, no. 13, pp. 3393–3406, Jul. 2018.

[6] M. Jian, F. Gao, Z. Tian, S. Jin, and S. Ma, "Angle-domain aided UL/DL channel estimation for wideband mmWave massive MIMO systems with beam squint," *IEEE Trans. Wireless Commun.*, to be published.

[7] R. W. Heath Jr., N. Gonzlez-Prelcic, S. Rangan, W. Roh, and A. M. Sayeed, "An overview of signal processing techniques for millimeter wave MIMO systems," *IEEE J. Sel. Topics in Signal Process.*, vol. 10, no. 3, pp. 436–453, Apr. 2016.

[8] L. Liang, W. Xu, and X. Dong, "Low-complexity hybrid precoding in massive multiuser MIMO systems," *IEEE Wireless Commun. Lett.*, vol. 3, no. 6, pp. 653–656, Dec. 2014.

[9] A. Alkhateeb, O. E. Ayach, G. Leus, and R. W. Heath Jr., "Channel estimation and hybrid precoding for millimeter wave cellular systems," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 831–846, Oct. 2014.

[10] Z. Gao, C. Hu, L. Dai, and Z. Wang, "Channel estimation for millimeter-wave massive MIMO with hybrid precoding over frequency-selective fading channels," *IEEE Commun. Lett.*, vol. 20, no. 6, pp. 1259–1262, Jun. 2016.

[11] Y. Wang, W. Xu, H. Zhang, and X. You, "Wideband mmWave channel estimation for hybrid massive MIMO with low-precision ADCs," *IEEE Wireless Commun. Lett.*, vol. 8, no. 1, pp. 285–288, Feb. 2019.

[12] Z. Zhou, J. Fang, L. Yang, H. Li, Z. Chen, and R. S. Blum, "Low-rank tensor decomposition-aided channel estimation for millimeter wave MIMO-OFDM systems," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 7, pp. 1524–1538, Jul. 2017.

[13] H. Ghauch, T. Kim, M. Bengtsson, and M. Skoglund, "Subspace estimation and decomposition for large millimeter-wave MIMO systems," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 3, pp. 528–542, Apr. 2016.

[14] Z. Qin, H. Ye, G. Y. Li, and B.-H. Juang, "Deep learning in physical layer communications," *IEEE Wireless Commun.*, vol. 26, no. 2, pp. 93–99, Apr. 2019.

[15] H. Ye, G. Y. Li, and B.-H. Juang, "Power of deep learning for channel estimation and signal detection in OFDM systems," *IEEE Wireless Commun. Lett.*, vol. 7, no. 1, pp. 114–117, Feb. 2018.

[16] H. He, C.-K. Wen, S. Jin, and G. Y. Li, "Deep learning-based channel estimation for beamspace mmWave massive MIMO systems," *IEEE Wireless Commun. Lett.*, vol. 7, no. 5, pp. 852–855, Oct. 2018.

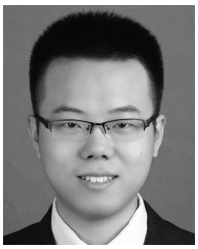
[17] C.-K. Wen, W.-T. Shih, and S. Jin, "Deep learning for massive MIMO CSI feedback," *IEEE Wireless Commun. Lett.*, vol. 7, no. 5, pp. 748–751, Oct. 2018.

[18] T. Wang, C.-K. Wen, S. Jin, and G. Y. Li, "Deep learning-based CSI feedback approach for time-varying massive MIMO channels," *IEEE Wireless Commun. Lett.*, vol. 8, no. 2, pp. 416–419, Apr. 2019.

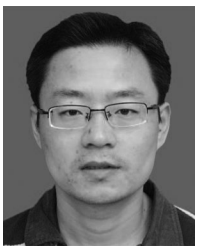
[19] H. He, S. Jin, C.-K. Wen, F. Gao, G. Y. Li, and Z. Xu, "Model-driven deep learning for physical layer communications," *IEEE Wireless Commun.*, to be published.

[20] J.-M. Kang, C.-J. Chun, and I.-M. Kim, "Deep-learning-based channel estimation for wireless energy transfer," *IEEE Commun. Lett.*, vol. 22, no. 11, pp. 2310–2313, Nov. 2018.

[21] P. Dong, H. Zhang, and G. Y. Li, "Machine learning prediction based CSI acquisition for FDD massive MIMO downlink," in *Proc. IEEE Global Commun. Conf.*, Abu Dhabi, United Arab Emirates, Dec. 2018, pp. 1–6.
[22] Y.-S. Jeon, S.-N. Hong, and N. Lee, "Blind detection for MIMO systems with low-resolution ADCs using supervised learning," in *Proc. IEEE Int. Conf. Commun.*, Paris, France, May 2017, pp. 1–6.
[23] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv: 1409.1556*, 2014.
[24] R. Hunger, "Floating point operations in matrix-vector calculus," Technische Universität München, Associate Institute for Signal Processing, Tech. Rep. v1.3, 2007.
[25] K. He and J. Sun, "Convolutional neural networks at constrained time cost," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Boston, MA, USA, Jun. 2015, pp. 5353–5360.
[26] V. Va, J. Choi, and R. W. Heath Jr., "The impact of beamwidth on temporal channel variation in vehicular channels and its implications," *IEEE Trans. Veh. Technol.*, vol. 66, no. 6, pp. 5014–5029, Jun. 2017.
[27] J. Choi, D. Love, and P. Bidigare, "Downlink training techniques for FDD massive MIMO systems: Open-loop and closed-loop training with memory," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 802–814, Oct. 2014.
[28] X. Shi, Z. Chen, H. Wang, D. Y. Yeung, W. K. Wong, and W. C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. Neural Inf. Process. Syst.*, 2015, pp. 802–810.
[29] 3GPP, "5G; Study on channel model for frequencies from 0.5 to 100 GHz," 3rd Generation Partnership Project (3GPP), Tech. Rep. 38.901 V15.0.0, Jun. 2018.



**Peihao Dong** (S'15) received the B.S. degree in communication engineering from Shandong University, Weihai, China, in 2012 and the M.S. degree in communication and information system from Shandong University, Jinan, China, in 2015. He is currently pursuing the Ph.D. degree at the National Mobile Communications Research Laboratory, Southeast University, Nanjing, China. Since October 2017, he has been a Visiting Student with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, USA. His current research interests include massive MIMO, machine learning for wireless communications and relaying transmission.



**Hua Zhang** (M'04) received the B.S. and M.S. degrees from the Department of Radio Engineering, Southeast University, Nanjing, China, in 1998 and 2001, respectively, and the Ph.D. degree from the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, USA, in 2004. From 2001 to 2004, he was a Graduate Research Assistant with the Georgia Institute of Technology. From 2004 to 2005, he was a Senior System Engineer with Skyworks Solutions Inc., Irvine, CA, USA. From 2005 to 2006, he was a Staff Engineer with MaxLinear Inc., Carlsbad, CA, USA. Since August 2006, he has been an Associate Professor with the School of Information Science and Engineering, Southeast University, Nanjing, China. Prof. Zhang was the recipient of the best paper awards of IEEE Microwave, Antenna, Propagation, and EMC Technologies in 2013, and IEEE Global Communication Conference in 2014. His current research interests include massive MIMO, software defined radio, and cooperative communications.

**Geoffrey Ye Li** (S'93–M'95–SM'97–F'06) received the B.S.E. and M.S.E. degrees from the Department of Wireless Engineering, Nanjing Institute of Technology, Nanjing, China, in 1983 and 1986, respectively, and the Ph.D. degree from the Department of Electrical Engineering, Auburn University, Auburn, AL, USA, in 1994.

He was a Teaching Assistant and then a Lecturer with Southeast University, Nanjing, China, from 1986 to 1991, a Research and Teaching Assistant with Auburn University from 1991 to 1994, and a Postdoctoral Research Associate with the University of Maryland, College Park, MD, USA, from 1994 to 1996. He was with AT&T Labs—Research at Red Bank, NJ, USA, as a Senior and then a Principal Technical Staff Member from 1996 to 2000. Since 2000, he has been with the School of Electrical and Computer Engineering, Georgia Institute of Technology as an Associate Professor and then a Full Professor. He has authored and coauthored more than 500 journals and conference papers in addition to over 40 granted patents on the topics of his research interests, which include statistical signal processing and machine learning for wireless communications. His publications have been cited over 35 000 times and he has been recognized as the World's Most Influential Scientific Mind, also known as a Highly-Cited Researcher, by Thomson Reuters almost every year. Prof. Li was the recipient of IEEE Fellow for his contributions to *Signal Processing for Wireless Communications* in 2005. He won the 2010 IEEE ComSoc Stephen O. Rice Prize Paper Award, 2013 IEEE VTS James Evans Avant Garde Award, 2014 IEEE VTS Jack Neubauer Memorial Award, 2017 IEEE ComSoc Award for Advances in Communication, and 2017 IEEE SPS Donald G. Fink Overview Paper Award. He also received 2015 Distinguished Faculty Achievement Award from the School of Electrical and Computer Engineering, Georgia Tech. He has been involved in editorial activities for over 20 technical journals for the IEEE, including Founding Editor-in-Chief of IEEE 5G Tech Focus. He has organized and chaired many international conferences, including Technical Program Vice-Chair of IEEE International Conference on Communications (2003), Technical Program Co-Chair of IEEE Signal Processing Advances in Wireless Communications (2011), General Chair of IEEE Signal and Information Processing (2014), Technical Program Co-Chair of IEEE Vehicular Technology Conference [2016 (Spring)], and General Co-Chair of IEEE Vehicular Technology Conference [2019 (Fall)].

**Ivan Simões Gaspar** received the Dr.-Ing. degree from Technische Universitt Dresden, Germany, in 2016, and the M.Sc. and B.S.E.E. degrees from Inatel, Brazil, in 2006 and 2003, respectively. From 2003 to 2011, he was a Technical Supervisor and Product Manager with the R&D Department, Hitachi Kokusai Linear. From 2008 to 2011, he collaborated as a Lecturer at Inatel. From 2012 to 2015, he was a Research Associate with the Vodafone Chair, focusing on the 5GNOW project, the NI Lead User Program, and pioneering the research of GFDM. Since 2016, he has been with Intel, Santa Clara, CA, USA, working on future wireless communication systems with great interest on advancing the science and technology of intelligent machines.

**Navid NaderiAlizadeh** received the B.S. degree in electrical engineering from the Sharif University of Technology, Tehran, Iran, in 2011; the M.S. degree in electrical and computer engineering from Cornell University, Ithaca, NY, USA, in 2014; and the Ph.D. degree in electrical engineering from the University of Southern California, Los Angeles, CA, USA, in 2016. He is currently a Research Scientist at Intel Labs, Santa Clara, CA, USA. His research interests include development and analysis of model-based and learning-based radio resource allocation algorithms for 5G and beyond. Dr. Naderializadeh ranked first in the Iranian Nationwide University entrance exam in 2007. He was the recipient of Jacobs Scholarship in 2011. He was selected as a 2015-16 Ming Hsieh Institute Ph.D. Scholar. He has also been a finalist in the Shannon Centennial Student Competition at Nokia Bell Labs in 2016.

# 毕业论文（设计）文献综述和开题报告考核

对文献综述、外文翻译和开题报告评语及成绩评定

该同学结合毕业设计课题，查阅了有关文献，外文翻译比较准确，文献综述基本反映了国内外研究现状及研究热点和难点。

根据毕业设计的要求，开题报告提出了基于 AI 信道估计的初步设计想法，基本明确预期达到的目标，安排了毕业设计时间进度。答辩导师组听取了该同学的开题报告，并就有关问题进行询问。经过讨论，一致通过开题报告，同意进入下一阶段毕设工作，该同学对研究内容和方案的理解还不够清晰，后续需抓紧明确和细化。

| 成绩比例 | 文献综述<br>占（10%） | 开题报告<br>占（15%） | 外文翻译<br>占（5%） |
|---|---|---|---|
| 分 值 | 8 | 12 | 4 |

开题报告答辩小组负责人（签名）_____

年　　月　　日