

CSE5010 期末报告——基于毫米波雷达的手写轨迹追踪

12332152 吉辰卿 12010508 华羽霄

1. 项目背景

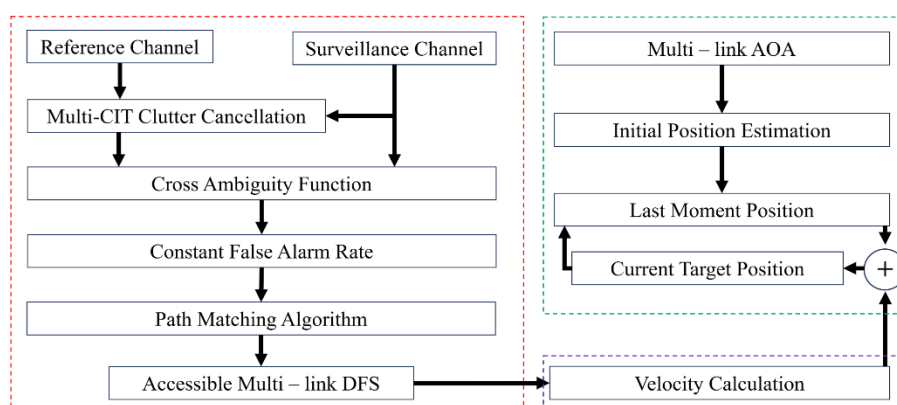
在人体运动跟踪中，已经有许多利用无线通信信道状态信息（CSI）进行估计的工作。然而，传感性能受到信号波长的限制。此外，基于 CSI 的传感方法对接收信号强度具有很高的灵敏度。这是因为精确估计 CSI 对于捕获手指运动引起的相移至关重要^[1, 2, 3]。

因此，许多现有作品中的手写跟踪实验都是在靠近 WiFi 发射机和接收机的地方进行的，它们之间有视线（LoS）路径。在非视距链路下，由于散射和反射的损失导致接收信号功率显著下降，同时，CSI 易受载波频偏（CFO）、采样频偏（SFO）和包检测时延（PDD）的影响导致其相位提取易受干扰，精确相位的提取显得较为困难。

在本项目中，我们采用毫米波雷达作为通信方式，考虑到其具有宽带宽和高分辨率等优秀特性。

2. 项目简介

在室内环境中，搭建毫米波雷达感知系统，重构书写运动轨迹。系统框图如下：



通过多链路毫米波被动雷达获取物体不同方位的多普勒信息，通过多普勒与物体真实速度之间的转换，从而实现手写轨迹的重构。通过交叉模糊函数、CFAR 和路径匹配算法处理接受信号，来实现低发射信号要求下对物体运动多普勒的提取。

3. 应用场景

该系统有如下应用场景。

- 手势识别与控制：通过分析人体运动的微小变化，可以实现手势识别和手势控制，例如在智能家居中控制灯光、电器等。
- 健康监测：跟踪人体运动可以用于健康监测，例如监测呼吸、心率等生理信号，有助于提供非侵入性的健康监护。
- VR 和 AR：在虚拟或增强现实环境中，可以使用这项技术来更自然地追踪用户的头部和身体运动，提高用户体验。
- 安防系统：可用于建立智能监控系统，非侵入地监测区域内的活动，实现对异常行为的实时检测。

4. 技术优势

毫米波手写追踪相较于 WiFi 手写追踪在一些方面可能具有优势，特别是在以下几个角度：

- 宽带宽：毫米波通信通常在毫米波频段工作，具有更大的带宽。这使得它能够传输更多数据，有助于提高手写追踪的精度和分辨率。
- 提高分辨率：由于宽带特性，毫米波系统可能具有更高的分辨率，能够更准确地捕捉手写迹线的细节。
- 高数据速率：由于宽带宽，毫米波通信通常能够提供更高的数据传输速率，具有较低的传播延迟。
- 方向性好：毫米波用得上定向天线，集中把功率打到一个方向减少功率损耗。
- 小天线：毫米波频段的天线尺寸相对较小，这有助于设计更紧凑、轻便的手写追踪设备。

5. 设备布置

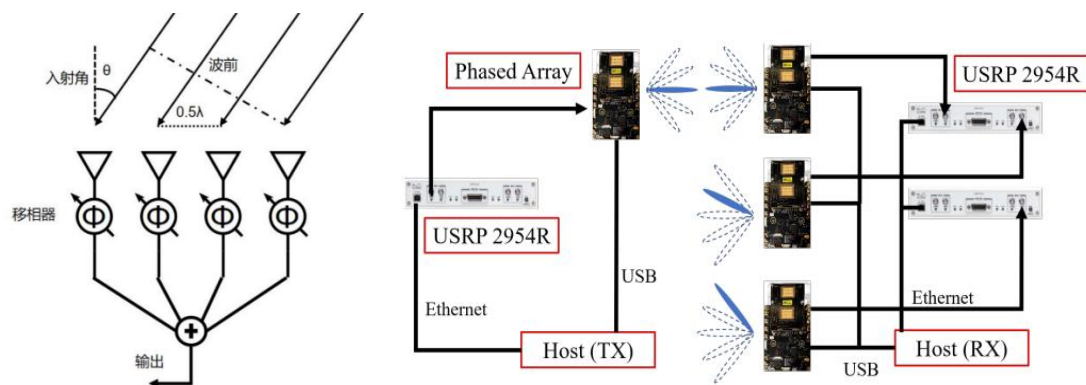
对于手写轨迹重构问题，使用四台 Sivers 60 GHz 相控阵列天线。设备参数如下：

- RX 采样率：1M
- 射频端中心频率：60.48 GHz
- 中频端中心频率：500 MHz

对于每一台天线，都有 16 个天线单元，在发射端取第 i 种波束，接收端取第 j 种波束时，波束赋形的信号^[4]为：

$$y(t) = w_j^H H v_i s(t) + n(t)$$

通过开启不同数量的天线单元和取不同的波束模式，可以对相控阵列天线的波束宽窄和波束方向进行调整。通过开启指定数量的天线单元和波束模式，实现对指定方向的信号获取。相控阵天线示意图如下：



在 4 台天线中，其中一台作为发射天线，因为实验是在通信感知一体化的背景下进行，发射的信号模拟真实的通信信号，在 IEEE802.11 标准中，信号通过 OFDM 方案进行调制，将发送的 OFDM 信号定义为：

$$s(t) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} a_n e^{\frac{j2\pi n t}{T}}$$

一台天线作为参考天线，参考天线接收的信号表示为：

$$y_r(t) = \sum_{i=1}^{L_r} \alpha_r^i s(t - \tau_r^i) e^{-j2\pi \Delta t} + n_r(t)$$

因为直达径的功率远大于其他路径，参考天线接收的信号表示为：

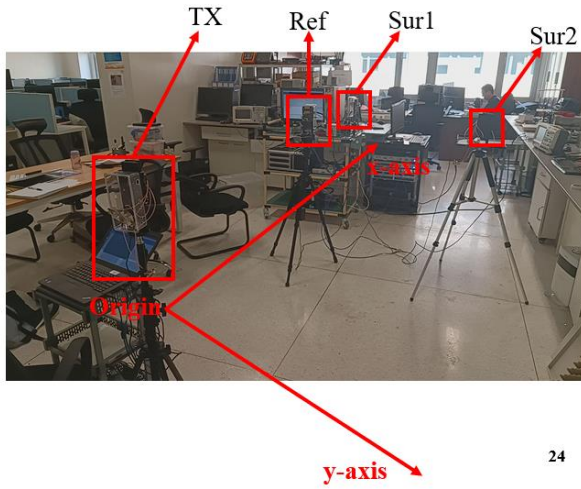
$$y_{r,i}(t) = h_{r,i} s(t - \tau_{r,i}) + n_{r,i}(t), \quad 0 \leq t \leq T$$

另外两台作为监视天线，监视天线接受的信号表示为：

$$y_{s,i}(t) = h_{s,i}^{\text{tar}}(t) s(t - \tau_{s,i}^{\text{tar}}(t)) e^{-j2\pi f_{d,i}^{\text{tar}}(t)t} + \sum_{l=1}^{L_i} h_{s,i}^l s(t - \tau_{s,i}^l) + n_{s,i}(t), \quad 0 \leq t \leq T$$

注意：i 表示感测时隙，我们有两个监视信道。每个监视信道的信号模型是相同的。

设备摆放位置如下：



坐标参数（米）		
发射天线	xT	0
	yT	0
监视天线 1	xR1	3
	yR1	0
监视天线 2	xR2	sqrt(2)
	yR2	-sqrt(2)
手写区域	xTar	2.5
	yTar	-1.5

24

6. 技术细节

6.1. 杂波消除

在交叉模糊函数后，存在较强的零频分量^[5]，这是来源于直达径的信号影响，因此需要做杂波消除。

具体使用基于最小二乘的自适应杂波消除算法来消除静态环境反射信号的干扰^[6]。

V_r 是参考信道的延迟参考信号矩阵：

$$\mathbf{V}_r = \begin{bmatrix} y_r[1] & y_r[0] & \cdots & y_r[-P+2] \\ y_r[2] & y_r[1] & \cdots & y_r[-P+3] \\ \vdots & \vdots & \vdots & \vdots \\ y_r[N] & y_r[N-1] & \cdots & y_r[N-P+1] \end{bmatrix}$$

\mathbf{V}_s 是监视信道的矩阵：

$$\mathbf{V}_s = [y[1], y[2], \cdots, y[N]]^T$$

用最小二乘法拟合延迟系数：

$$\mathbf{K} = (\mathbf{V}_r^H \mathbf{V}_r)^{-1} \mathbf{V}_r^H \mathbf{V}_s$$

基于最小二乘法杂波消除后的监视信道信号：

$$\mathbf{y}_s^* = \mathbf{y}_s - \mathbf{V}_r \mathbf{K}$$

在本项目中，我们在相干积分时间（CIT）内执行杂波消除算法。CIT 表示滑动窗口的长度。在实际计算时，为了避免零频的完全去除，选择采用多 CIT 杂波消除的方式。

6.2. 交叉模糊函数

在被动雷达中，对参考信道和监视信道进行同步后，使用交叉模糊函数处理来获得多普勒偏移。通过估计从参考信道和监视信道收集的信号之间的相关性，CAF 能够估计目标参数，如时延 τ 和多普勒频移 f_d 。

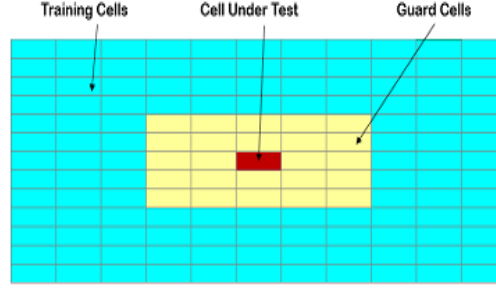
滑动窗口+CAF 将获得时间-多普勒频谱图，表示为：

$$R(f_d) = \max_{\tau} \sum_{n=0}^{N-1} y_s[n] \{y_r[n - \tau]\}^\dagger \cdot e^{-j2\pi f_d n T_s}$$

可以观察到，当时延和多普勒匹配时，参考信号和监视信号的模糊函数计算值会达到峰值，通过遍历的方式可以得到峰值对应的多普勒频移，在实际运算中，采用快速傅里叶变换来加快运算速度。

6.3. 恒定虚警概率下的检测器

对于两个监视信道得到的两幅时间多普勒图，因为毫米波对微多普勒的敏感性，会存在一些与手写轨迹多普勒无关的噪声，因此采用恒定虚警概率下的检测器（CFAR）进行处理。



在本项目中，我们采用基于自适应阈值的方法来检测 CAF 中的多普勒频率。这就是 2D-CFAR 算法^[7]。在 CFAR 中，有一些关键组件：

- 检测单元：需要与阈值进行比较的单元称为检测单元（CUT）。
- 训练单位：CUT 周围等数量的单元用于计算背景噪声功率，这些单元称为训练单位。在本次项目的实验中，我们选择**训练带大小=5**。
- 防护单位：CUT 和训练单位之间有几个单位不参与背景噪声功率的计算，这些单位称为保护单位。在本次项目的实验中，我们选择**保护带大小=5**。

2D-CFAR 算法对的关键参数：

- 检测阈值： $\beta = \alpha * P_n$ 。它由背景噪声功率 P_n 和阈值因子 α 决定。

- 背景噪声功率： $P_n = \frac{1}{N} \sum_{m=1}^N x_m$ 。其中 N 为训练单元数量。

- 虚警概率与门限因子： $a = N (P_{fa}^{-1/N} - 1)$ 。其中 a 为门限因子， f_a 为虚警概率。在本次项目的实验中，我们选择**误报率=0.7**。

在 CAF 结果之后应用 CFAR 算法，我们得到了更清晰的时间多普勒频谱。

6.4. 路径匹配算法

CFAR 算法不能完全消除假阳性，一些异常值仍会影响当前时刻的多普勒频率估计（因为当前感知时刻 CAF 最大值对应的多普勒频率被选为当前时刻的多普勒频率）。CFAR 算法也会产生一些漏检，导致多普勒频率值的一些时刻缺失。这将严重影响后续的速度和位置的迭代估计，因此需要设计路径匹配算法^[7, 8, 9]。

我们注意到物体的运动是连续的，也就是说，当前时刻的多普勒频移值应该与前一时刻比较接近，不会出现大的跳跃。设计路径算法时，主要考虑两种情况：

当前时刻无检测结果。采用滑动平均对当前时刻进行插值：

$$f_{d,i} = f_{d,before} + \frac{f_{d,after} - f_{d,before}}{\Delta t} (t_i - t_{before})$$

当前时间多普勒结果可用时。讨论两种情况：

- 当检测结果唯一时，选择唯一的检测结果作为当前时刻的多普勒。
- 当检测结果不唯一时，由于多普勒在时间前后具有连续性，首先考虑距离权重函数，并考虑多普勒的绝对值。因此，在选择点时定义了以下权重函数：

$$w_{i,j} = \alpha e^{-d_{i,j}^{before}} + \beta f_{d,i,j}$$

其中， d_{before} 表示当前时刻检测到的多普勒频率点与前一时刻之间的距离， f_d 表示当前时刻检测到的多普勒频率的绝对值。 α 和 β 都是我们需要调节的超参数。在本次项目的实验中，我们选择 $\alpha=10$ ， $\beta=1$ 。

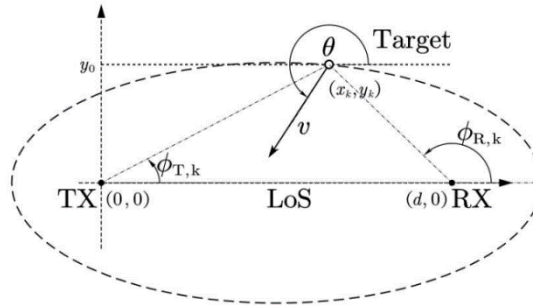
我们遍历当前检测到的所有多普勒值，并对每个多普勒值应用权重函数。最后，我们选择使权重函数值最大的多普勒频率值作为当前时刻对象运动的多普勒频率值。

6.5. 位置迭代模型

路径匹配后，可以分别获得两个方向的多普勒轨迹，但多普勒描述的是反射路径变换的速度，并不等于物体运动的真实速度。对于单个链路，多普勒与速度的真实大小之间存在关系^[10]：

$$f_D = -\frac{f_{TX}}{c} \cdot v \cos(\theta - \frac{\alpha_R + \alpha_T}{2}) \cdot 2 \cos(\frac{\alpha_R - \alpha_T}{2})$$

其中， α_R 是接收器的 AoA， α_T 是发射器的 AoD， θ 是目标的运动方向。因此，从上面的方程可以看出，如果我们想根据多普勒频率获得目标当前运动的速度和方向，我们需要一个二元方程来求解这两个未知数。



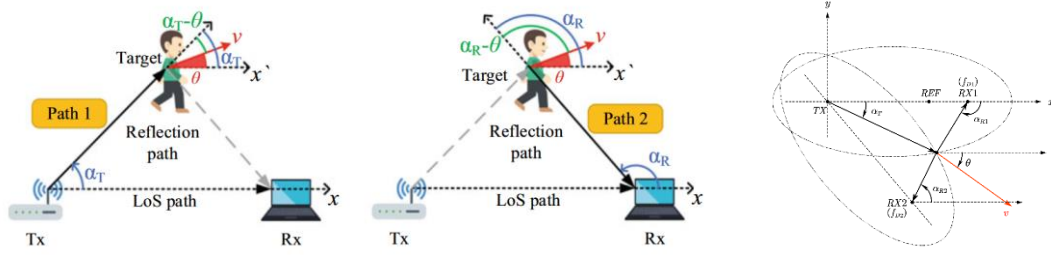
但是，如果想要解出速度的大小和方向，需要多链路的多普勒，建立如下方程：

$$\begin{cases} f_{D1} = -\frac{f_o}{c} \cdot v \cos\left(\theta - \frac{\alpha_T + \alpha_{R1}}{2}\right) \cdot 2 \cos\frac{\alpha_T - \alpha_{R1}}{2}, \\ f_{D2} = -\frac{f_o}{c} \cdot v \cos\left(\theta - \frac{\alpha_T + \alpha_{R2}}{2}\right) \cdot 2 \cos\frac{\alpha_T - \alpha_{R2}}{2}. \end{cases}$$

其中， f_{D1} 为第一条链路的多普勒， f_{D2} 为第二条链路的多普勒。注意：在本项目中，我们假设发射机、两个监视信道的接收机以及物体的初始位置都是已知的。因此，在迭代的每

一步过程中，我们可以根据移动物体在之前的时刻的位置来获得 α_1 、 α_2 和 α_T 。

具体表示含义如图所示：



对方程组进行整理：

$$\begin{cases} f_{D1} = -\frac{f_0}{c} \cdot v \cos \theta \cos \left(\frac{\alpha_T + \alpha_{R1}}{2} \right) \cdot 2 \cos \left(\frac{\alpha_T - \alpha_{R1}}{2} \right) - \frac{f_0}{c} \cdot v \sin \theta \sin \left(\frac{\alpha_T + \alpha_{R1}}{2} \right) \cdot 2 \cos \left(\frac{\alpha_T - \alpha_{R1}}{2} \right) \\ f_{D2} = -\frac{f_0}{c} \cdot v \cos \theta \cos \left(\frac{\alpha_T + \alpha_{R2}}{2} \right) \cdot 2 \cos \left(\frac{\alpha_T - \alpha_{R2}}{2} \right) - \frac{f_0}{c} \cdot v \sin \theta \sin \left(\frac{\alpha_T + \alpha_{R2}}{2} \right) \cdot 2 \cos \left(\frac{\alpha_T - \alpha_{R2}}{2} \right) \end{cases}$$

把速度在 x 和 y 方向分解的分量 v_x 和 v_y 代入方程，可以表示为：

$$\begin{cases} f_{D1} = -\frac{f_0}{c} \cdot v_x \cos \left(\frac{\alpha_T + \alpha_{R1}}{2} \right) \cdot 2 \cos \left(\frac{\alpha_T - \alpha_{R1}}{2} \right) - \frac{f_0}{c} \cdot v_y \sin \left(\frac{\alpha_T + \alpha_{R1}}{2} \right) \cdot 2 \cos \left(\frac{\alpha_T - \alpha_{R1}}{2} \right) \\ f_{D2} = -\frac{f_0}{c} \cdot v_x \cos \left(\frac{\alpha_T + \alpha_{R2}}{2} \right) \cdot 2 \cos \left(\frac{\alpha_T - \alpha_{R2}}{2} \right) - \frac{f_0}{c} \cdot v_y \sin \left(\frac{\alpha_T + \alpha_{R2}}{2} \right) \cdot 2 \cos \left(\frac{\alpha_T - \alpha_{R2}}{2} \right) \end{cases}$$

表示为矩阵形式，并对速度分量进行求解，可以得到：

$$\begin{bmatrix} v_x \\ v_y \end{bmatrix} = \begin{bmatrix} -\frac{f_0}{c} \cdot \cos \left(\frac{\alpha_T + \alpha_{R1}}{2} \right) \cdot 2 \cos \left(\frac{\alpha_T - \alpha_{R1}}{2} \right) & -\frac{f_0}{c} \cdot \sin \left(\frac{\alpha_T + \alpha_{R1}}{2} \right) \cdot 2 \cos \left(\frac{\alpha_T - \alpha_{R1}}{2} \right) \\ -\frac{f_0}{c} \cdot \cos \left(\frac{\alpha_T + \alpha_{R2}}{2} \right) \cdot 2 \cos \left(\frac{\alpha_T - \alpha_{R2}}{2} \right) & -\frac{f_0}{c} \cdot \sin \left(\frac{\alpha_T + \alpha_{R2}}{2} \right) \cdot 2 \cos \left(\frac{\alpha_T - \alpha_{R2}}{2} \right) \end{bmatrix}^{-1} \begin{bmatrix} f_{D1} \\ f_{D2} \end{bmatrix}$$

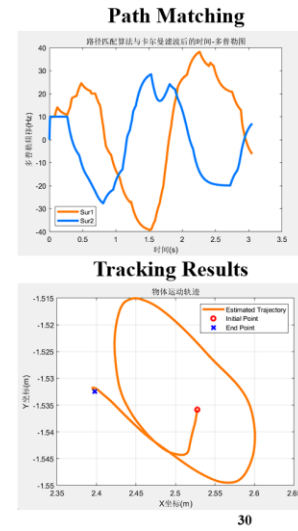
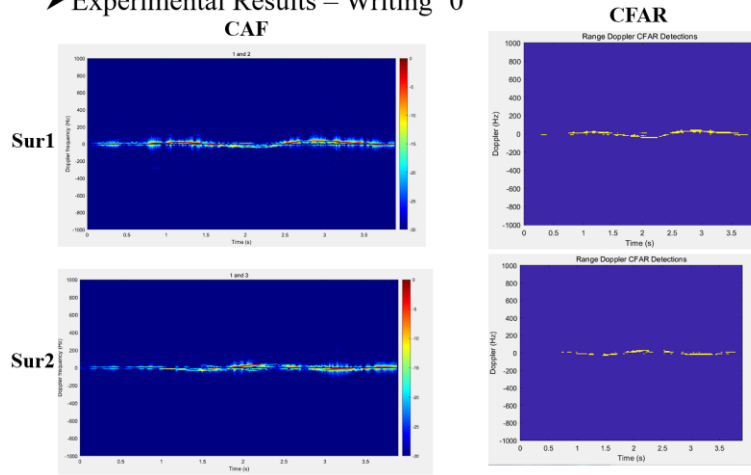
因此，由两幅时间多普勒图可以得到物体真实速度在 x, y 轴上的速度分量，对于每一时刻的速度分量，乘以相干积分时间，可以还原物体的轨迹。

$$\begin{cases} x_{k+1} = x_k + v_k N_0 T_s \cos \theta_k \\ y_{k+1} = y_k + v_k N_0 T_s \sin \theta_k \end{cases}$$

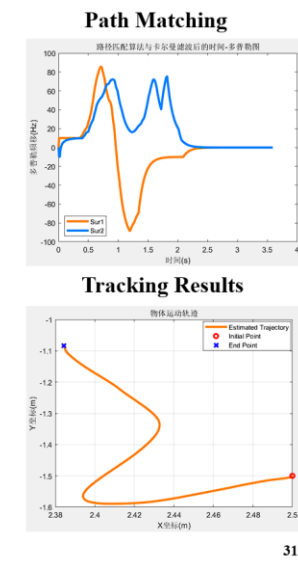
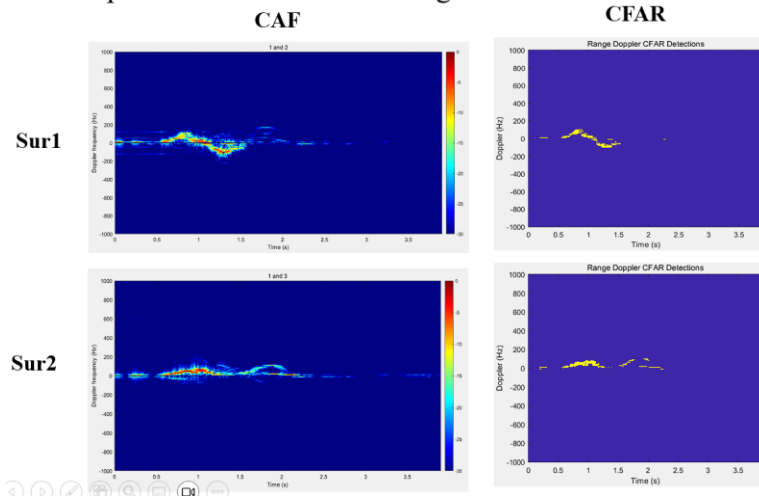
7. 实验结果

7.1. 整体结果

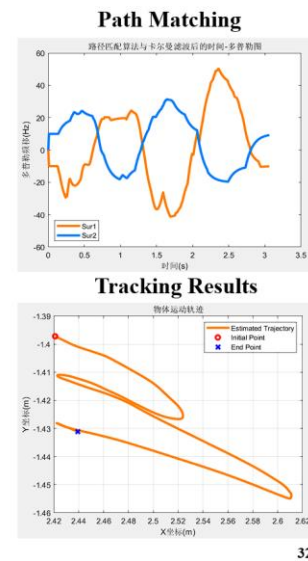
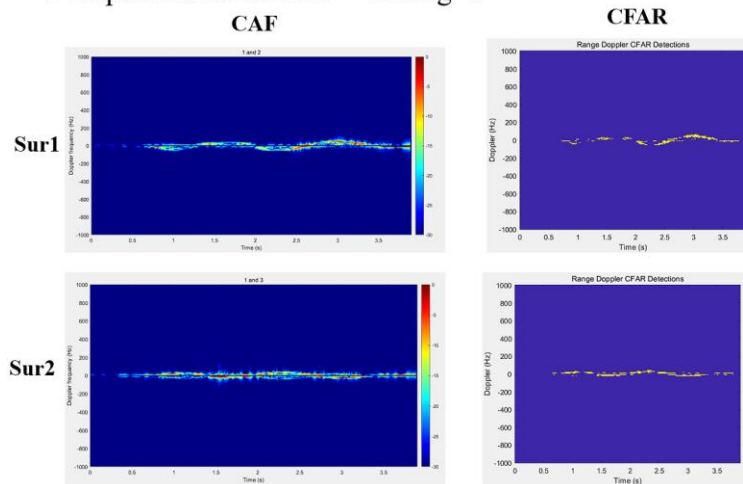
➤ Experimental Results – Writing ‘0’



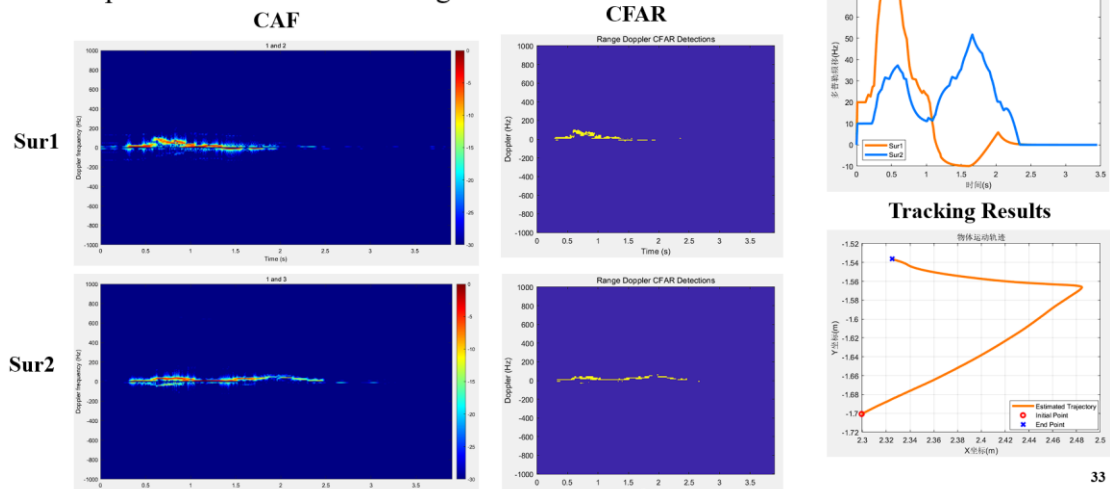
➤ Experimental Results – Writing ‘2’



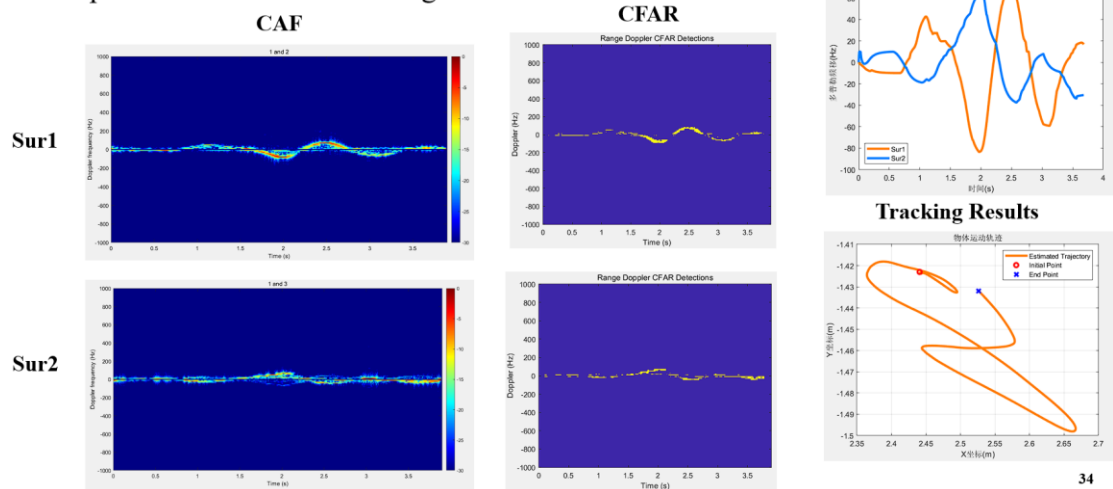
➤ Experimental Results – Writing ‘3’



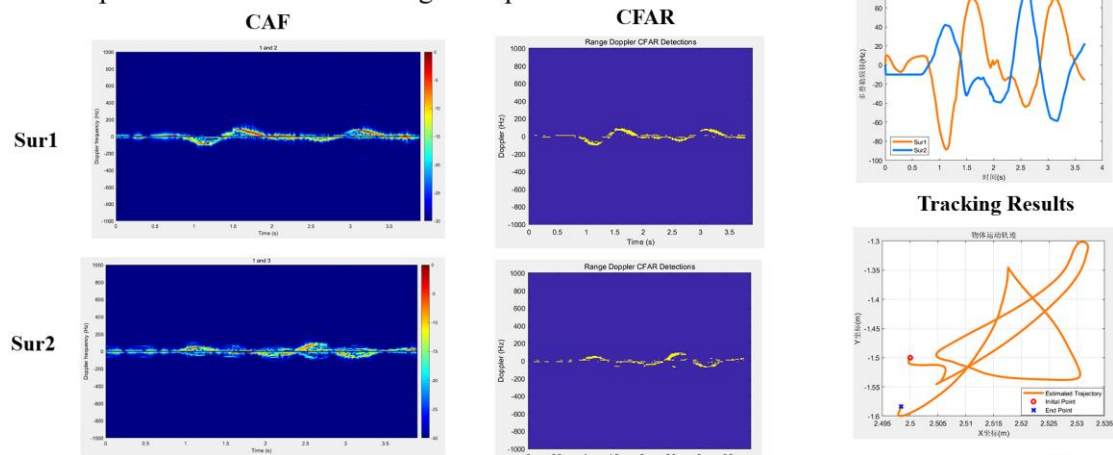
➤ Experimental Results – Writing ‘7’



➤ Experimental Results – Writing ‘8’

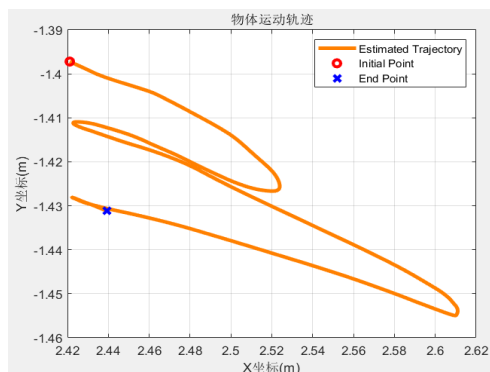
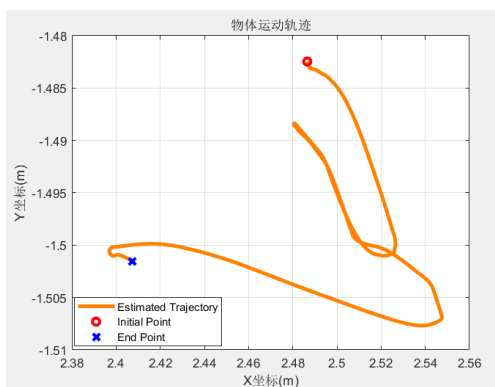
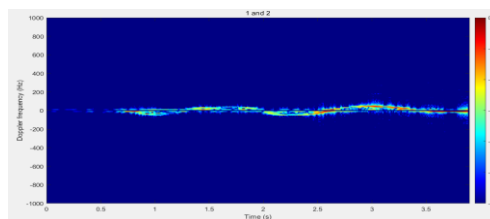
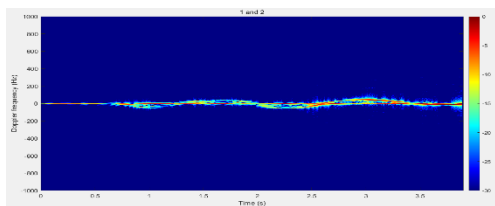


➤ Experimental Results – Writing ‘five-pointed star’



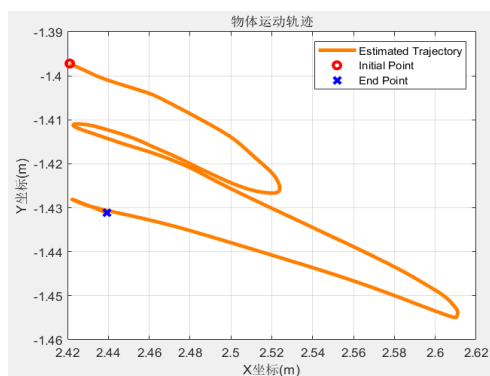
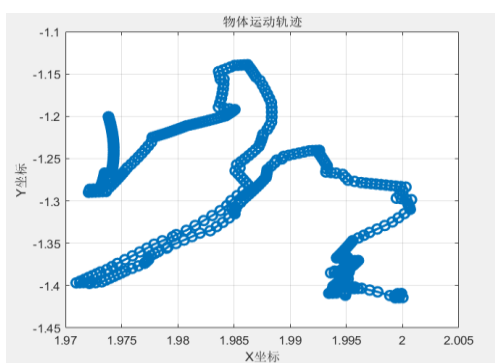
7.2. 消融实验

7.2.1. 杂波消除效果前后对比



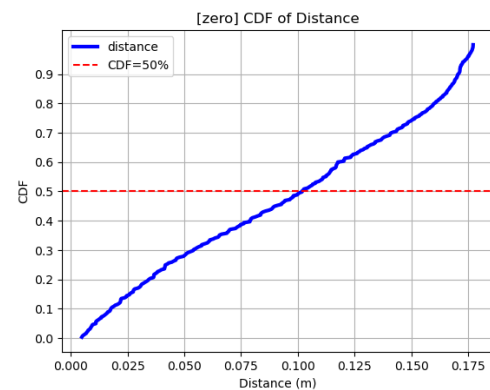
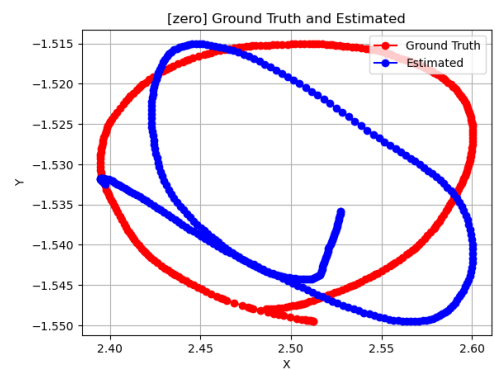
可以看到，杂波消除后，零频附近的信息大部分被丢弃，书写轨迹跟接近真实情况。

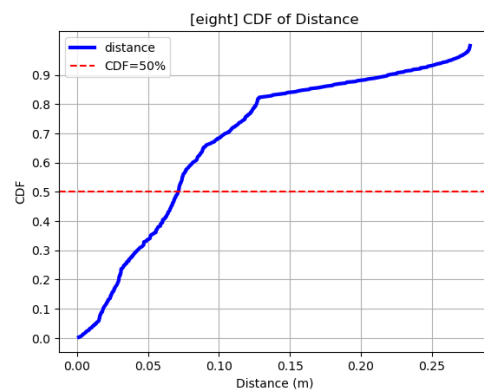
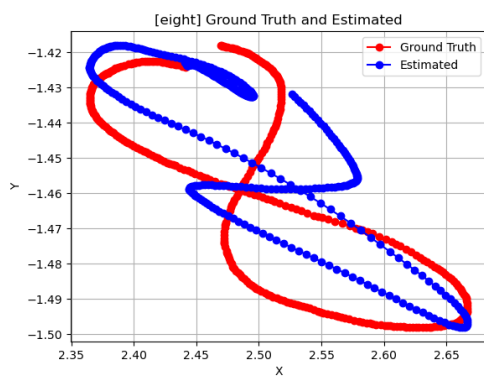
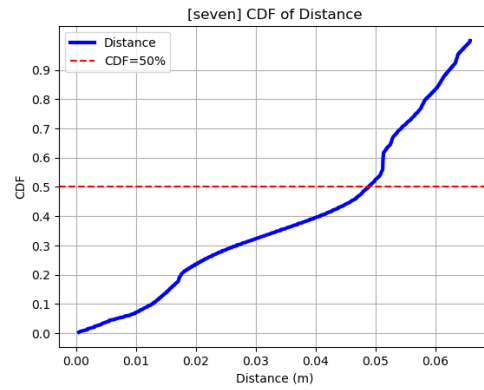
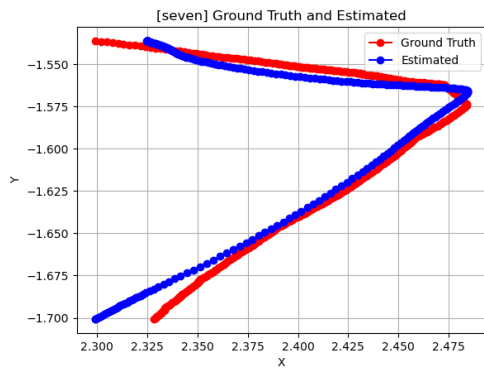
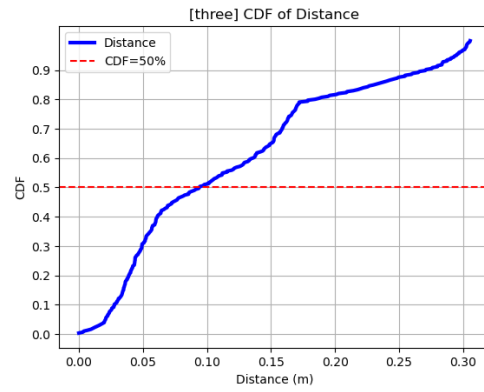
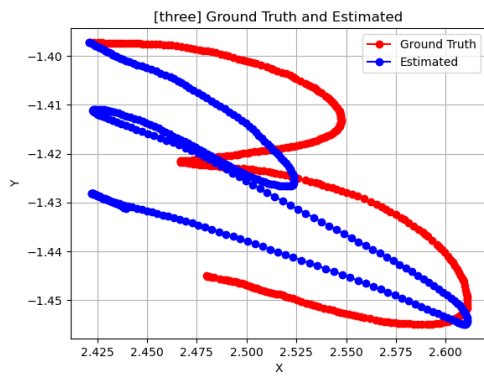
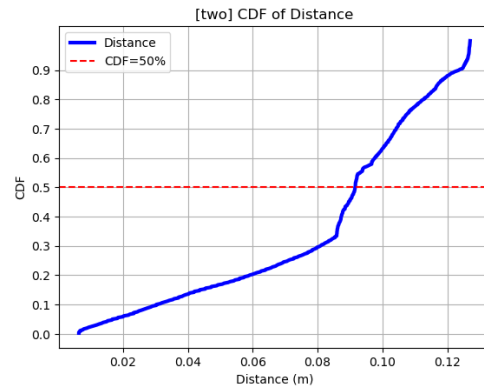
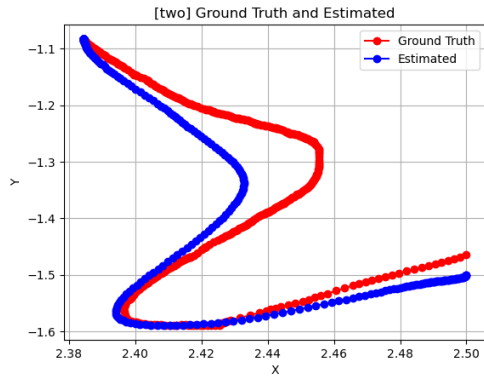
7.2.2. CFAR 和路径匹配效果前后对比

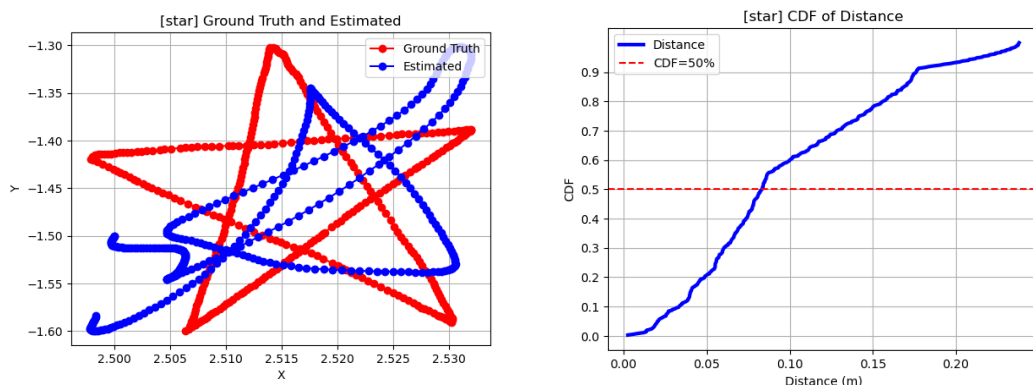


初始 CAF 结果会产生大量错误检测、漏检和异常值，这将导致对象运动的多普勒估计不准确。恢复的轨迹将非常不均匀（异常多普勒值跳跃太大）。

7.3. 轨迹重构及误差分布







可以看出，这六种手写笔迹的跟踪与 GroundTruth 之间的 50%误差 CDF（中值误差）在 10cm 以内。

8. 总结思考

在该项目中，我们设计搭建了一个基于毫米波雷达的手写轨迹追踪系统。系统由一个毫米波相控阵发射机和三个毫米波相控阵接收机（作为参考信道、监视信道 1 和监视信道 2）组成。

我们提取运动目标两个方向的多普勒频率分量，采用自主研发的路径匹配算法解决了 CFAR 算法后的多普勒频率缺失、异常值和错误检测等问题。

实验结果证明，路径匹配算法在准确、流畅地恢复运动物体的轨迹方面起着重要作用。通过写入 5 个不同的数字和五角星图案，证明我们提出的系统可以准确地还原真实轨迹，中值误差基本在厘米级。

9. 未来计划

9.1. 多人手写追踪

由于路径匹配算法更适合检测每个人的连续多普勒频率，因此我们可以提取在时-多普勒频谱图中分别提取多个人的连续多普勒，并对它们进行区分。

首先，通过聚类算法分离出同一个时间-多普勒图中不同人书写轨迹的时间-多普勒图像。接着，分别用路径匹配还原每一个人的真实书写轨迹。然后，调整路径匹配算法的超参数来追踪多人的手写，从而验证该算法是否对多人手写追踪保持了良好的准确性。

9.2. 三维手写追踪

目前系统只能得到多普勒频率在 xy 平面方向的投影，可以实现同一水平面上地轨迹估计。如果要想实现空间内的轨迹估计，考虑到高度不同，我们的任务变成了三维情况下的手写追踪，多普勒分量多了 z 轴的分量。因此我们还要在手写区域正上方加一条新的监视信道链路朝下（称为监视信道 3）。

在讨论手写轨迹时，发现写小尺度的三、八等小字时效果很好，但在写大尺度的轨迹时，尤其是画大型的三或其他图形时，恢复效果不佳。这问题可能源于分辨率不足，因此考虑通过调整相干积分时间（ C 值）来改变分辨率，以实现在更大尺度的手势运动下准确还原轨迹。

9.3. 大尺度运动

在讨论手写轨迹时，我们发现恢复小尺度的 3、8 等效果很好，但在恢复大尺度的轨迹

时，尤其是画大型的 3 或其他图形时，恢复效果不佳。

这问题可能源于分辨率不足。由于 CIT 值的导数即多普勒分辨率，当前 CIT 值为 0.1，导致小尺度运动的多普勒分辨率最高为 10Hz。对于大尺度运动（如手臂、衣服等晃动），可能存在较大的干扰，需要更高的分辨率。增大 CIT 值可以提高多普勒分辨率。小尺度运动目前 CIT 值足够，但对于大尺度运动，可能需要增大 CIT 值以提高分辨率，剔除混杂干扰。

```
1. CIT = 0.1;  
2. CIT_region = CIT*f_s;  
3. N_slide = 10;  
4. T_slide = CIT / N_slide;  
5. max_dop = 1000;  
6. step_dop = 1/CIT;  
7. array_Doppler_frequency = -max_dop:step_dop:max_dop;
```

因此考虑通过调整相干积分时间（CIT 值）来改变分辨率，以实现在更大尺度的手势运动下准确还原轨迹。

参考文献

- [1] L. Sun, S. Sen, D. Koutsonikolas, and K.-H. Kim, "Widraw: Enabling hands-free drawing in the air on commodity wifi devices," in Proceedings of the 21st Annual International Conference on Mobile Computing and Networking, 2015, pp. 77–89.
- [2] Z. Han, Z. Lu, X. Wen, W. Zheng, J. Zhao, and L. Guo, "Centitrack: Towards centimeter-level passive gesture tracking with commodity wifi," IEEE Internet of Things Journal, 2023.
- [3] L. Wang, K. Sun, H. Dai, A. X. Liu, and X. Wang, "Witrace: Centimeterlevel passive gesture tracking using wifi signals," in 2018 15th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON). IEEE, 2018, pp. 1–9.
- [4] Wu K, Chen R, Wang H, et al. Human Respiration Detection Under Interference: Challenges and Solutions[J]. arXiv preprint arXiv:2310.03297, 2023.
- [5] Li, J., Yu, C., Luo, Y., Sun, Y., & Wang, R. (2022). Passive Motion Detection via mmWave Communication System.
- [6] D. K. Tan, H. Sun, Y. Lu, M. Lesturgie, and H. L. Chan, "Passive radar using global system for mobile communication signal: theory, implementation and measurements," IEE Proceedings-Radar, Sonar and Navigation, vol. 152, no. 3, pp. 116–123, 2005.
- [7] R. Chen, "Indoor Human Trajectory Reconstruction via Multi-link mmWave Passive Sensing System," 2023 IEEE/CIC International Conference on Communications in China (ICCC), Dalian, China, 2023, pp. 1-6, doi: 10.1109/ICCC57788.2023.10233603.
- [8] Qian K, Wu C, Zhang Y, et al. Widar2. 0: Passive human tracking with a single Wi-Fi link[C]//Proceedings of the 16th annual international conference on mobile systems, applications, and services. 2018: 350-361.
- [9] Wu C, Zhang F, Wang B, et al. mmTrack: Passive multi-person localization using commodity millimeter wave radio[C]//IEEE INFOCOM 2020-IEEE Conference on Computer Communications. IEEE, 2020: 2400-2409.
- [10] Niu K, Wang X, Zhang F, et al. Rethinking Doppler effect for accurate velocity estimation

with commodity WiFi devices[J]. IEEE Journal on Selected Areas in Communications, 2022, 40(7): 2164-2178.

附录代码

杂波消除

```
1. for i= 1:length(array_start_time)-2
2.     %%%杂波消除
3.     tar_integer(i,:) = ClutterCancellation_Doppler(tar_integer(i,:),ref_integer(i,:));
4.     %%%杂波消除
5.     tar_integer2(i,:) = ClutterCancellation_Doppler(tar_integer2(i,:),ref_integer2(i,:));
6. end
```

CAF

```
1. for i= 1:length(array_start_time)-2
2.     %%%CAF
3.     final=fftshift(fft(tar_integer(i,:).*conj(ref_integer(i,:)),CIT_region));
4.     A_TD(i,:) = final(CIT_region/2+1-max_dop/step_dop:CIT_region/2+1+max_dop/step_dop);
5.     %%%CAF
6.     final2=fftshift(fft(tar_integer2(i,:).*conj(ref_integer2(i,:)),CIT_region));
7.     A_TD2(i,:) = final2(CIT_region/2+1-max_dop/step_dop:CIT_region/2+1+max_dop/step_dop);
8. end
```

CFAR

```
1. %提前消除 0 频最大多普勒分量
2. % plot_A_DT2(101, :) = -1000;
3. cfar2D = phased.CFARDetector2D('GuardBandSize',5,'TrainingBandSize',5,...
4.     'ProbabilityFalseAlarm',0.7);
5. resp=plot_A_DT2;
6. rngGrid=array_Doppler_frequency.';
7. dopGrid=array_start_time.';
8. rangeIndx(1)= 60;
9. rangeIndx(2)= 140;
10. dopplerIndx(1)= 11;
11. dopplerIndx(2)= 379;
12. [columnInds,rowInds] = meshgrid(dopplerIndx(1):dopplerIndx(2),...)
```

```

13. rangeIndx(1):rangeIndx(2));
14. CUTIdx = [rowInds(:) columnInds(:)]';
15. %CFAR 检测结果
16. detections_2 = cfar2D(resp,CUTIdx);
17. helperDetectionsMap(resp,rngGrid,dopGrid,rangeIndx,dopplerI
    ndx,detections_2)

```

路径匹配

```

1. %%path matching algorithm
2. endCol_first_1 = 0;
3. endCol_first_2 = 0;
4. resultDFList_1 = zeros(1,dopplerIndx(2)-dopplerIndx(1)+1);
5. resultDFList_2 = zeros(1,dopplerIndx(2)-dopplerIndx(1)+1);
6. % Extract and process 11st column
7. lastCol = Map_1(:,dopplerIndx(1));
8. lastCol_2 = Map_2(:,dopplerIndx(1));
9. nonZeroRow = find(lastCol ~= 0);
10. nonZeroRow_2 = find(lastCol_2 ~= 0);
11. %Weight parameter when there are multiple Doppler frequenci
    es
12. alpha_1 = 10;
13. beta_1 = 1;
14. alpha_2 = 10;
15. beta_2 = 1;
16. %%Find the first interpolation Doppler of the two channels
17. if isempty(nonZeroRow) % If the first moment is 0, you have
    to look later
18.     for i = dopplerIndx(1)+1 : dopplerIndx(2)
19.         testCol_first = Map_1(:,i);
20.         testNonZeroRow_first = find(testCol_first ~= 0)
            ;
21.         if isempty(testNonZeroRow_first)
22.             continue
23.         else
24.             endCol_first_1 = i;
25.             endRow_first_1 = mean(Row2Df(testNonZeroRow
                _first));
26.             break
27.         end
28.     end
29.     resultDF = endRow_first_1;
30.     resultDFList_1(endCol_first_1-
        dopplerIndx(1)+1) = resultDF;
31.     resultDFList_1(1:endCol_first_1-dopplerIndx(1)) = [];

```



```

32.else
33.    endCol_first_1 = dopplerIndx(1);
34.    resultDF = mean(Row2Df(nonZeroRow));
35.    resultDFList_1(1) = resultDF;
36.end
37.
38.% Extract and process endCol_first+1 st to 379th columns fo
   r channel 1
39.endCol_1 = 0;
40.for i = endCol_first_1+1 : dopplerIndx(2) %12 to 379
41.    if i < endCol_1+1
42.        continue
43.    end
44.
45.    thisCol = Map_1(:,i);
46.    nonZeroRow = find(thisCol ~= 0);
47.    if isempty(nonZeroRow)
48.        startCol = i;
49.        % Find forward until there exists a unempty column
50.        for j = i+1 : dopplerIndx(2)
51.            testCol = Map_1(:,j);
52.            testNonZeroRow = find(testCol ~= 0);
53.            if isempty(testNonZeroRow)
54.                continue
55.            else
56.                endCol_1 = j-1;
57.                endRow = mean(Row2Df(testNonZeroRow));
58.                break
59.            end
60.        end
61.        % Assign a value for sequence,
62.        % with index from startCol to endCol,
63.        % and value from resultRow to endRow
64.        slope = (endRow-resultDF)/((endCol_1-
            startCol+2)*T_slide);
65.        for k = startCol:endCol_1
66.            resultDFList_1(k-
                endCol_first_1+1) = resultDF + slope*((k-
                    startCol+1)*T_slide);
67.        end
68.
69.    elseif length(nonZeroRow) == 1 %Just one
70.        % Simply take it
71.        resultDF = Row2Df(nonZeroRow);

```

```

72.         resultDFList_1(i-endCol_first_1+1) = resultDF;
73.     else %There's multiple Dopplers
74.         % Find the nearest index
75.         distance = abs(Row2Df(nonZeroRow) - resultDF); % distance
76.         tempolate = alpha_1 * exp(-
            distance) + beta_1 * abs(Row2Df(nonZeroRow));
77.         [miniRow, miniCol] = find(tempolate==max(tempolate)
            );
78.         resultDF = Row2Df(nonZeroRow(miniRow(1)));
79.         resultDFList_1(i-endCol_first_1+1) = resultDF;
80.     end
81.end
82.
83.%如果两路信道的多普勒矩阵没有对齐，就要让他们对齐
84.[aligned_DF1, aligned_DF2] = align_matrices(resultDFList_1,
    resultDFList_2);

```

卡尔曼滤波

```

1. function smoothed_data = KalmanSmoother(input_data)
2.     A = 1; %
3.     H = 1; %
4.     Q = 1e-4; %
5.     R = 1e-2; %
6.
7.     x_hat = input_data(1); %
8.     P = 1; %
9.
10.    for k = 2:length(input_data)
11.        %
12.        x_hat_minus = A * x_hat;
13.        P_minus = A * P * A' + Q;
14.
15.        K = P_minus * H' / (H * P_minus * H' + R);
16.        x_hat = x_hat_minus + K * (input_data(k) - H * x_hat_minus);
17.        P = (1 - K * H) * P_minus;
18.
19.        smoothed_data(k) = x_hat;
20.    end
21.end
22.

```

位置迭代

```

1. %开始解方程与迭代
2. for i = 1:length(sensing_time)-2
3.     fd = [maxDF1(i);maxDF2(i)];
4.     F = -
        2*fc/c*[cos((fai_sur1(i) - fai_tx(i))/2)*cos((fai_sur1(i) +
        fai_tx(i))/2) , ...
5.         cos((fai_sur1(i) - fai_tx(i))/2)*sin((fai_sur1(i) +
        fai_tx(i))/2); ...
6.         cos((fai_sur2(i) - fai_tx(i))/2)*cos((fai_sur2(i) +
        fai_tx(i))/2), ...
7.         cos((fai_sur2(i) - fai_tx(i))/2)*sin((fai_sur2(i) +
        fai_tx(i))/2)];
8.     v_xy(i,:) = F \ fd;
9.     %判断解是否正确，如果解出来离谱的速度值，就要用之前的值进行加
        权平均
10.    % if abs(v_xy(i,1)) > 0.6
11.    %     v_xy(i,1) = (v_xy(i-1,1) + v_xy(i-2,1) + v_xy(i-
        3,1))/3;
12.    % elseif abs(v_xy(i,2)) > 0.6
13.    %     v_xy(i,2) = (v_xy(i-1,2) + v_xy(i-2,2) + v_xy(i-
        3,2))/3;
14.    % end
15.    xtar(i+1) = xtar(i) + v_xy(i,1) * T_slide;
16.    ytar(i+1) = ytar(i) + v_xy(i,2) * T_slide;
17.    %更新下一时刻的 AOA 角度以便下一次进行迭代
18.    fai_sur1(i+1) = atan((ytar(i+1))/(xtar(i+1) - xR1));
19.    fai_sur2(i+1) = atan((ytar(i+1) - yR2)/(xtar(i+1) - xR
        2));
20.    fai_tx(i+1) = atan(ytar(i+1)/xtar(i+1));
21. end

```