



北京理工大学

Beijing Institute of Technology

雷达数据智能处理 以手势识别任务为例

2025.7.25

手势识别

手势识别概述

□ 手势识别

◆ 跟踪人类手势、识别其表示和转换为语义上有意义的命令

接触式传感器



数据手套[3][4]

非接触类传感器



微型雷达[13]



无标记[14]



惯性传感器[6]



触摸屏[7]



标视觉深度[11][12]



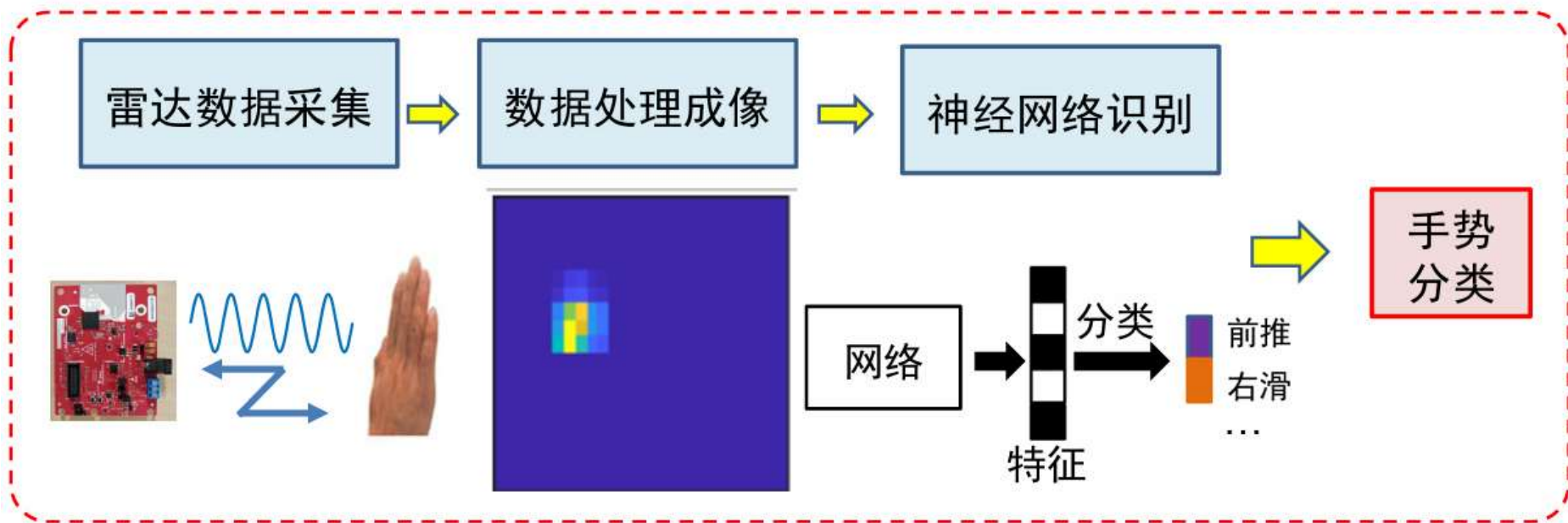
颜色标记[1]

手势识别

手势识别概述

□ 动态手势识别

- ◆ 数据处理：提取表征不同手势的信息，包括距离、速度、角度
- ◆ 神经网络识别与分类：设计网络，逐层提取特征，预测手势类别



手势识别

手势识别概述

- 相比可见光图像，雷达信号具有如下优点
 - ◆ 对光照条件鲁棒
 - ◆ 可以直接提取速度、距离等信息

手势识别

实验平台

□ 德州仪器AWR1642 单芯片

载频	77GHz	采样点数	256
带宽	4GHz	调频斜率	105.22MHz/ μ s
中频采样率	10MHz	天线收发方式	2发4收
ADC位数	16bit	采样形势	I/Q正交采样

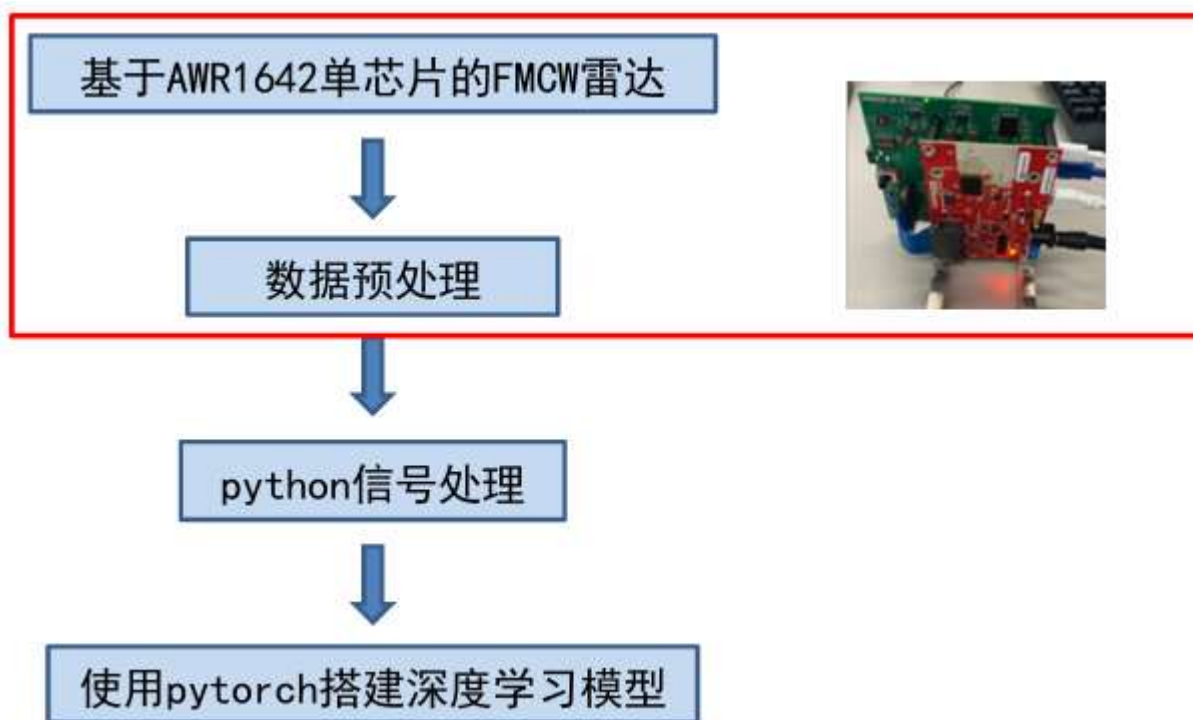


手势识别

基本流程

□ 信号采集与预处理

◆ 雷达通过手势动作进行信号捕捉并将数据转化为可处理格式



手势识别

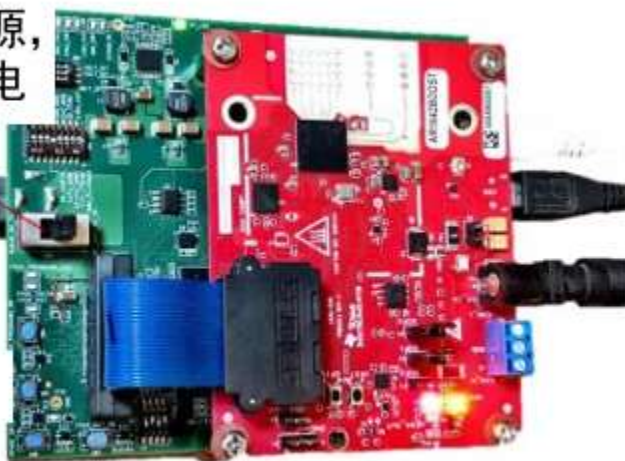
基本流程

□ 开发板连接方式

◆ 共计连接USB*2、RJ45网线*1、电源线*1

向右是外接电源，
向左是雷达供电

外接电源接口



跳线帽接上面两个SOP0和SOP1



DCA1000EVM
调试端口J1

AWR1642调试
端口J8

网线连接
DCA1000

雷达电源
5V2.5A

手势识别

基本流程

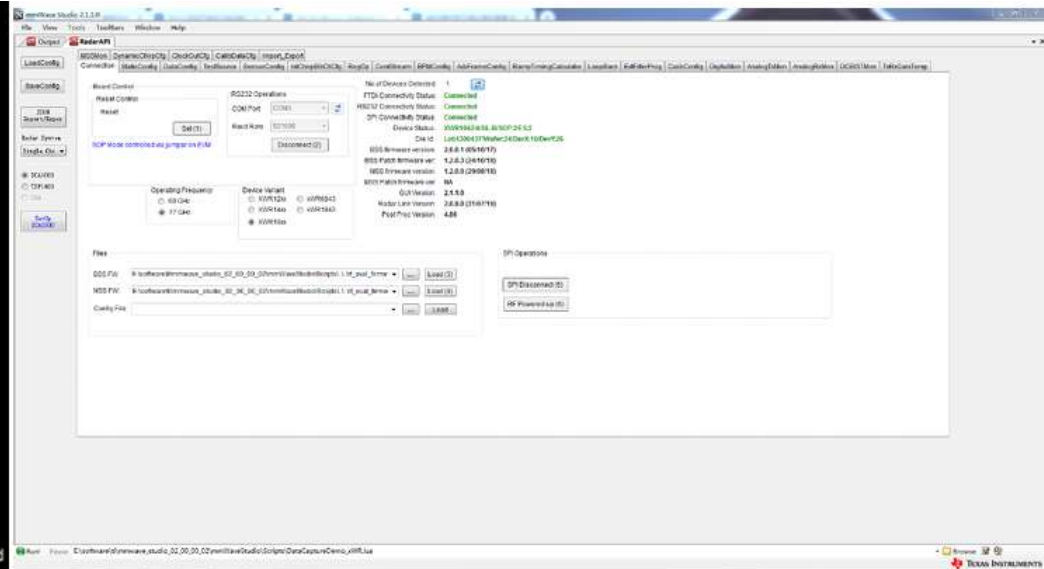
❑ Mmwavestudio上位机软件

- ◆ 直接获取基带信号、自带简易的分析处理程序
- ◆ 操作细节详见附录1

XWR1x mmWaveStudio
EVALUATION



Copyright 2016, Texas Instruments Incorporated. All rights reserved.



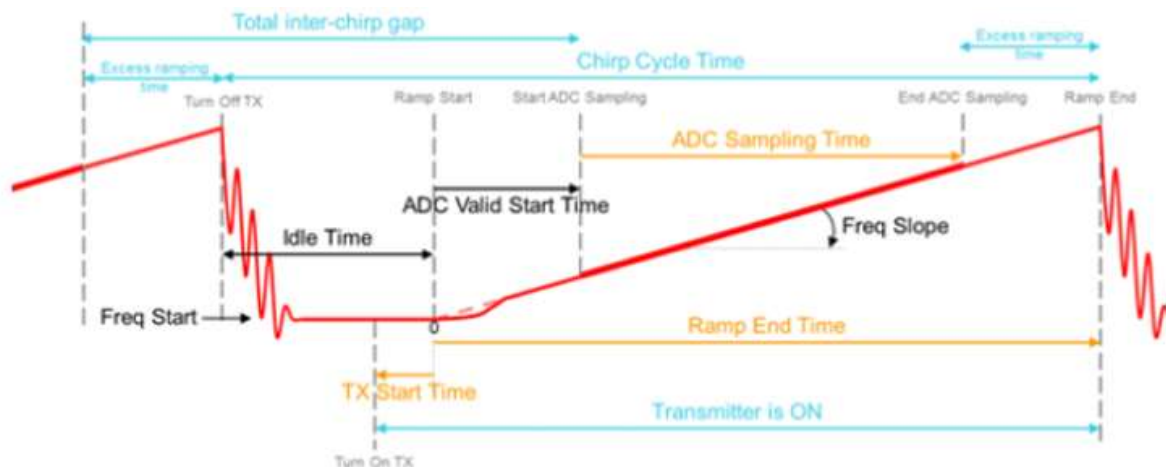
手势识别

基本流程

□ 可自行设置需要的参数，其他参数由系统自动计算

◆ 发射信号需要自行设置的参数

- Freq Slope: 线性调频信号的调频率
- Ramp End Time: 扫频时间



BLUE = Not a register. Shown for information only

BLACK = Fully configurable per chirp (through the chirp configuration RAM)

ORANGE = Configurable per chirp to one of 4 values, one per Chirp Profile

受限于硬件，chirp之间会有一段等待时间，期间没有信号发射。

只有图中ADC Sampling Time对应时间采样，等效采集到的数据为调频连续波采样数据

手势识别

基本流程

□ 采集要求

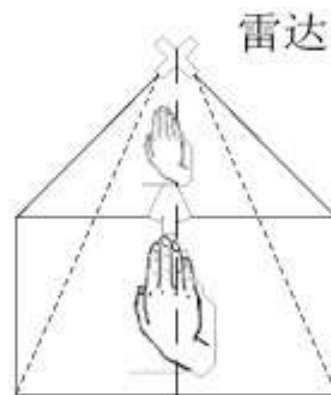
- ◆ 数据采集：4类手势
- ◆ 手势与雷达天线距离约20-40 cm
- ◆ 手势动作总时长：0.5-1 S



A



B



C



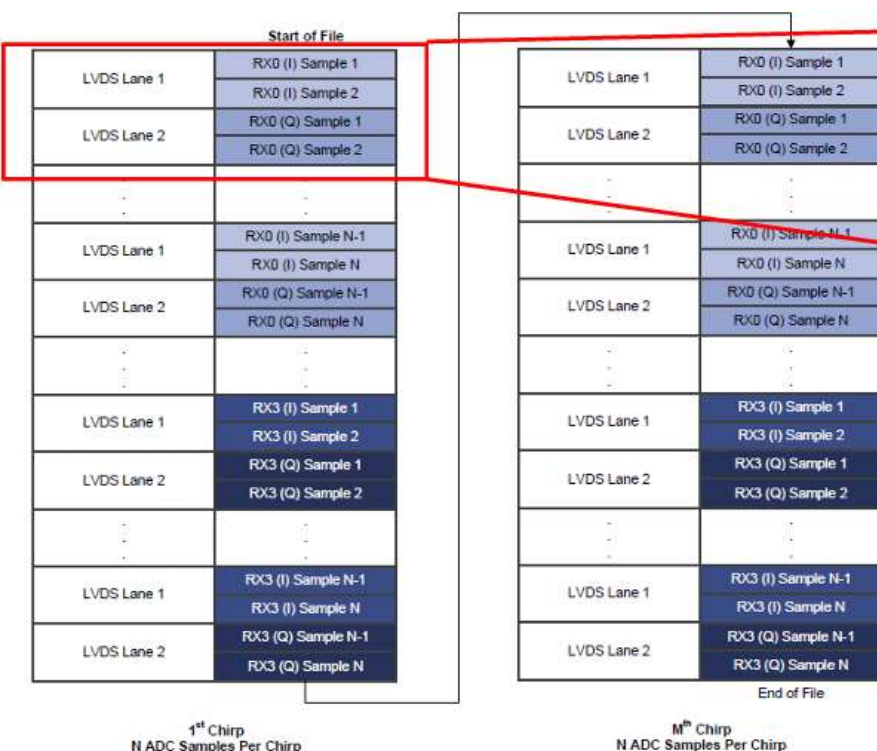
D

手势识别

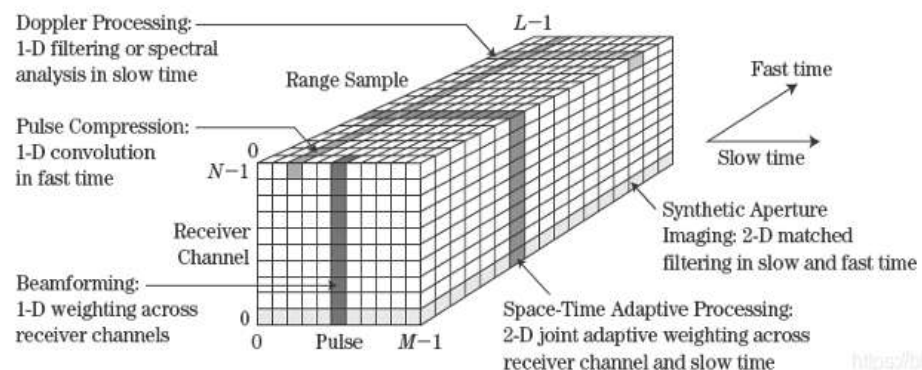
基本流程

□ 经DCA1000EVM数采板采集到的数据为中频信号采样

◆ 原始数据为二进制文件，采样位宽为16bit



对于每个chirp:
Lane1采集信号实部, Lane2采集
信号虚部, 每两个采样交替排列;
多接收天线采样数据按顺序排列



原始采集数据排列方式：一维

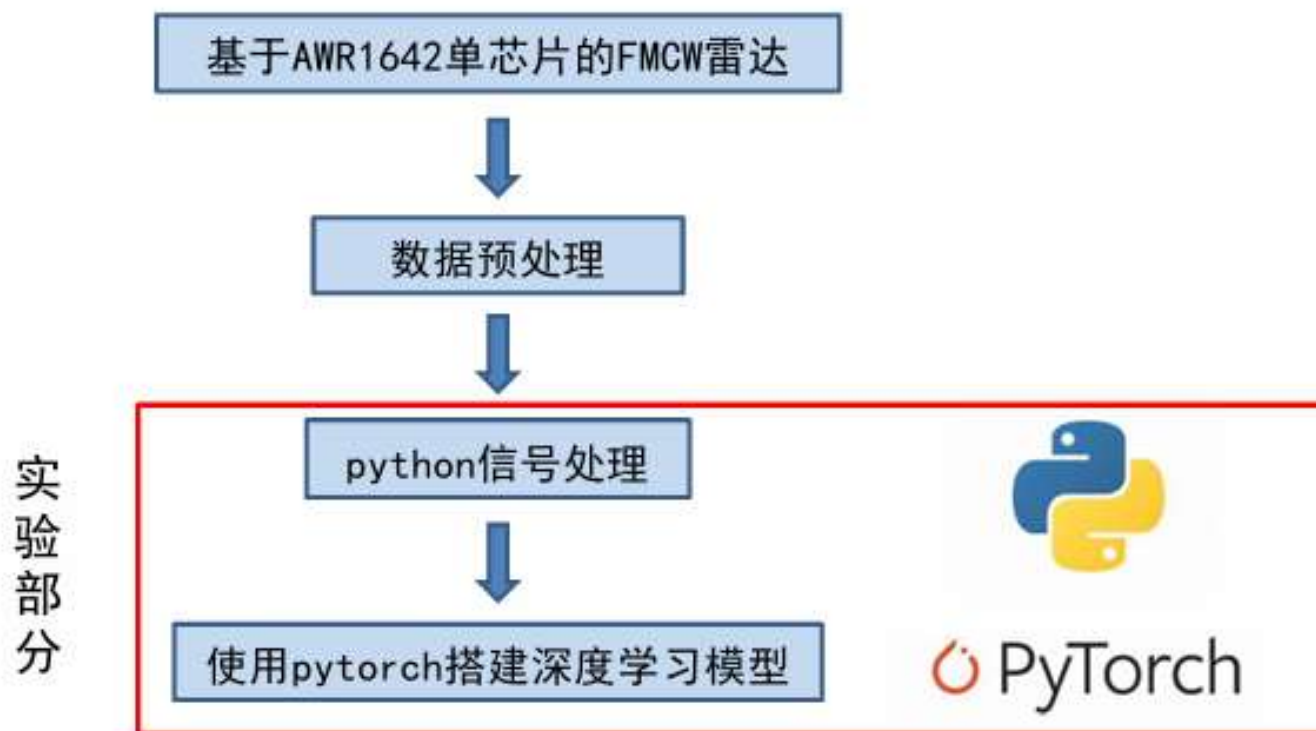
整理后数据排列方式：三维

手势识别

基本流程

□ 信号处理与网络识别

◆ 原始数据为二进制文件，采样位宽为16bit



手势识别

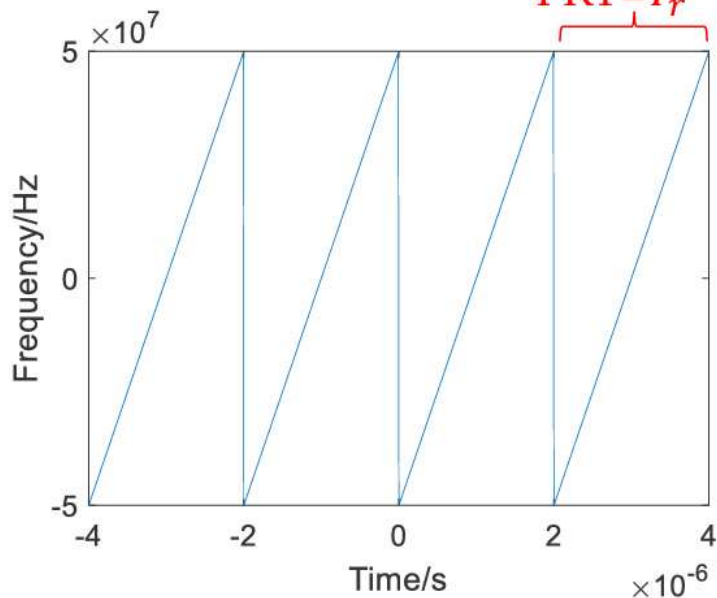
基本流程

□ 发射信号：线性调频连续波

◆ 具有与线性调频脉冲信号相同的频率调制形式

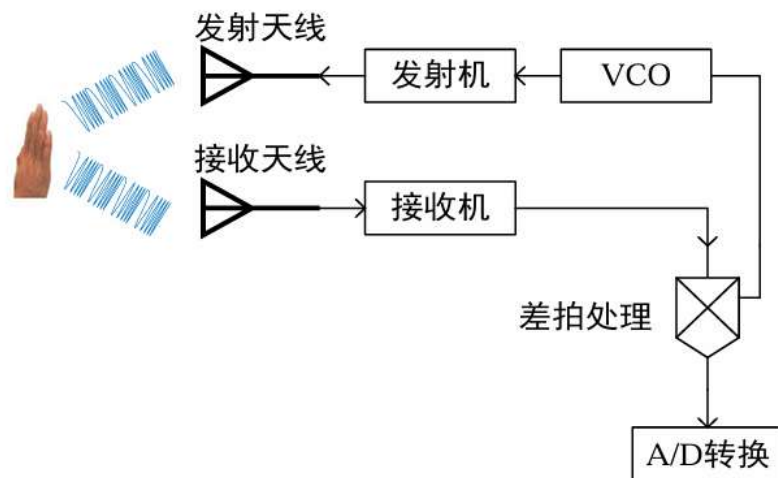
$$s(t) = \sum_{n=-\infty}^{+\infty} s_1(t - nT_r)$$

PRT = T_r



线性调频连续波时频图

$$s_1(t) = \begin{cases} \exp\left[j\pi \frac{B}{T_r} \left(t - \frac{T_r}{2}\right)^2\right], & 0 \leq t < T_r \\ 0 & \text{else} \end{cases}$$



线性调频连续波雷达系统框图

手势识别

基本流程

□ 数据预处理

- ◆ 将原始二进制数据转化为可编程使用的数据
- ◆ bin文件→mat文件

□ 信号处理

- ◆ 提取能有效表征不同手势的信息
 - 速度、距离、角度等时序变化

□ 网络识别

- ◆ 基于信号处理后的信息，转换成神经网络可以输入的形式
- ◆ 设计神经网络结构，逐层提取特征，并预测手势类别

手势识别

基本流程

□ 基本解决思路（只是一种方法，可以探索更多的其他方法）

◆ 信号处理

- 通过信号处理，计算手势的距离、速度和角度
- 利用距离、速度和角度三维参数在时间上的累积，将一个手势动作映射为固定帧数（比如32）距离速度-矩阵图（RD）和角度时间图（ATM）

◆ 网络识别

- 构建不同网络分别对于距离速度-矩阵图和角度时间图进行手势特征提取并进行特征融合

手势识别信号处理

信号处理流程

□ 手势数据的采集参数

- ◆ 1个手势动作：帧数=32
- ◆ 1帧：chirp数量=64
- ◆ 1个chirp：采样点=256

采样点数设置

ADC Samples

256

Chirp数量设置

No of Chirp Loops

64

帧数设置

No of Frames

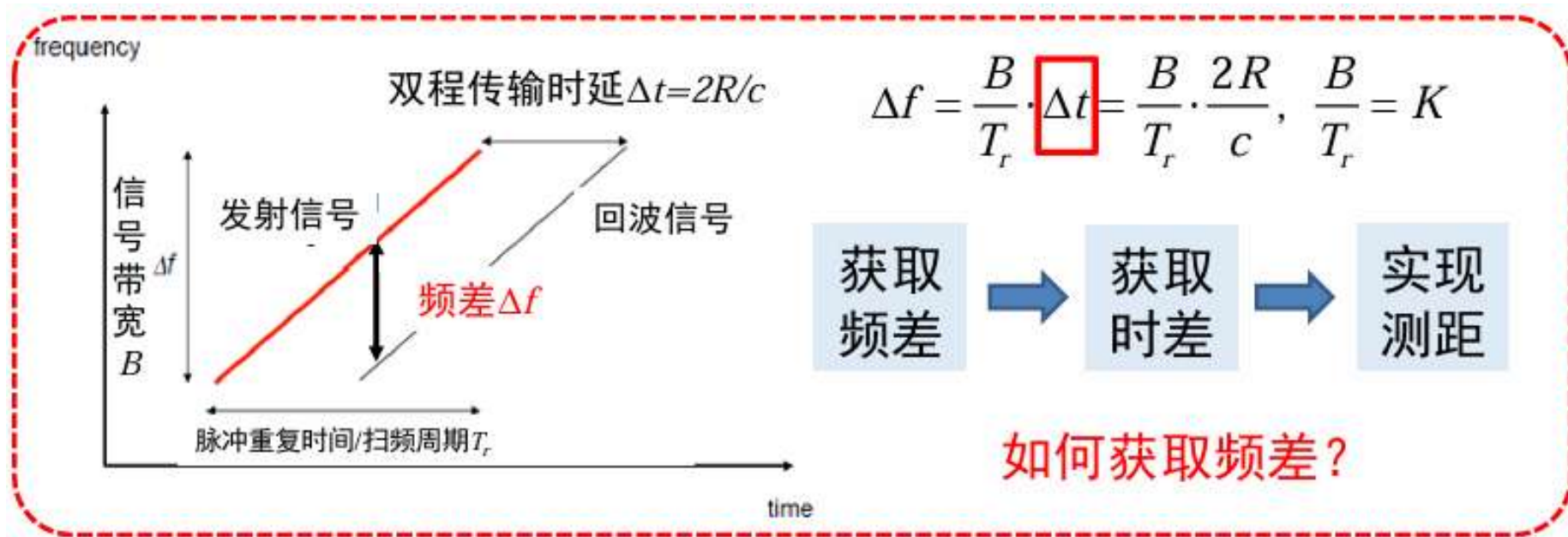
32

手势识别信号处理

信号处理流程

□ 线性调频连续波的脉冲压缩：去斜（Dechirp）

◆ Bin文件已经是中频信号采样，可直接用于下一步处理



手势识别信号处理

信号处理流程

□ 距离、速度信息提取：二维FFT

对静止目标，有

$$S_{TX}(t) = A_{TX} \exp\{j[2\pi f_0 t + kt^2]\} \quad (k = \frac{B}{2T})$$

$$S_{RX}(t) = A_{RX} \exp\{j[2\pi f_0(t - t_d) + k(t - t_d)^2]\}$$

$$S_{IF}(t) = A_{IF} \exp\{j2\pi f_0 t_d - j\pi k t_d^2\} \exp(j2\pi k t_d t)$$

快时间fft $\rightarrow f_s$

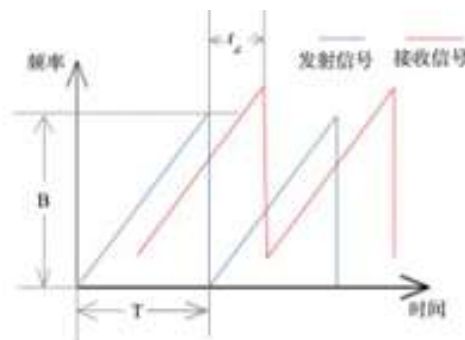


图1 发射信号与接收信号的时间相关曲线

对运动目标，有

$$S_{IF1}(t) = A_{IF1} \exp\{j2\pi f_0 t_d - j\pi k t_d^2\} \exp(j2\pi k t_d t) \exp(j2\pi f_d t_{PRT})$$

$$S_{IF2}(t) = A_{IF2} \exp\{j2\pi f_0 t_d - j\pi k t_d^2\} \exp(j2\pi k t_d t) \exp(j2\pi f_d * 2 * t_{PRT})$$

$$S_{IFn}(t) = A_{IFn} \exp\{j2\pi f_0 t_d - j\pi k t_d^2\} \exp(j2\pi k t_d t) \exp(j2\pi f_d * n * t_{PRT})$$

常数项

慢时间fft $\rightarrow f_d$

$$\text{手势到雷达的距离: } d = \frac{c f_s}{2k}$$

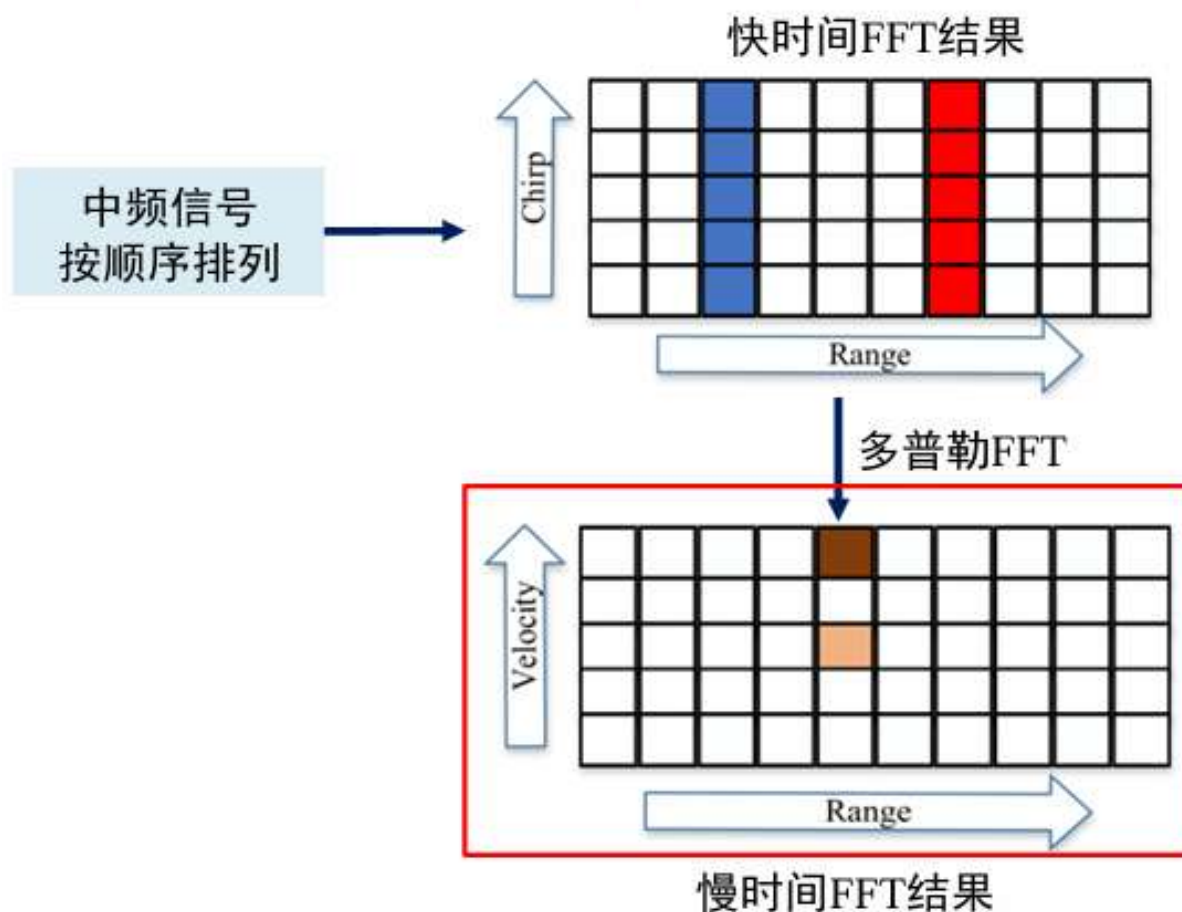
$$\text{手势运动速度: } v = \frac{\lambda f_d}{2}$$

手势识别信号处理

信号处理流程

□ 距离、速度信息提取：二维FFT

◆ 具体实现：二维FFT



- 获得横坐标为距离、纵坐标为速度的二维图像
- 目标对应的特性会在图中表现为对应位置的点

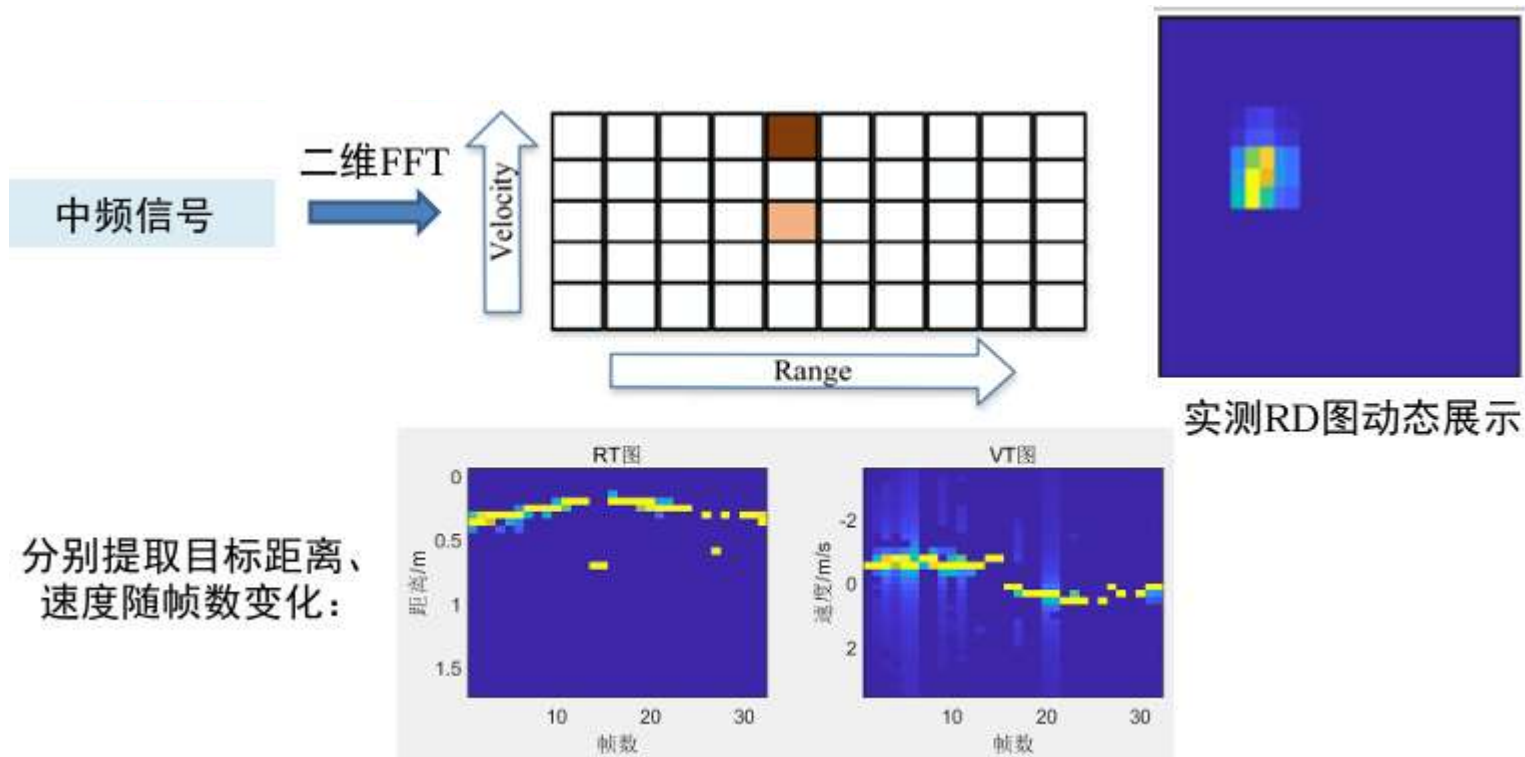
手势识别信号处理

信号处理流程

□ 距离、速度信息表现形式：RD图

◆ 横坐标：目标距离

◆ 纵坐标：目标速度



手势识别信号处理

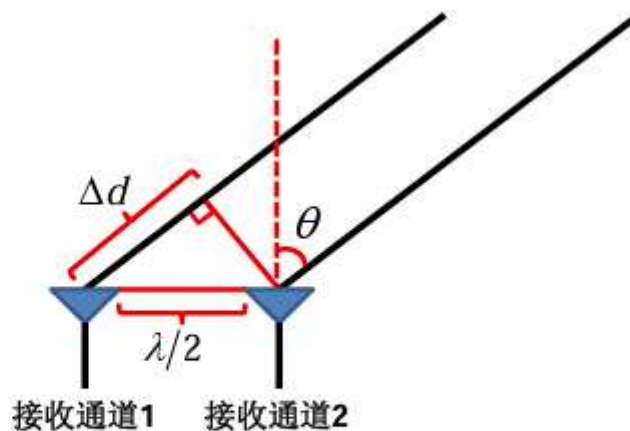
信号处理流程

□ 角度信息提取：利用天线的方向性

◆ 相位-角度敏感特性：基于远场假设

➤ 假设同一目标到达不同天线的回波互相平行

$$\Delta\varphi = \frac{2\pi\Delta d}{\lambda}$$



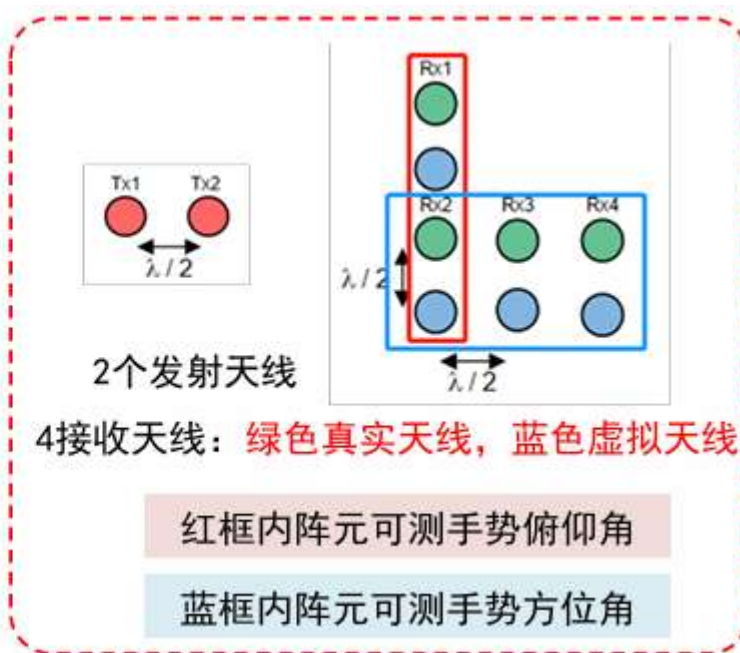
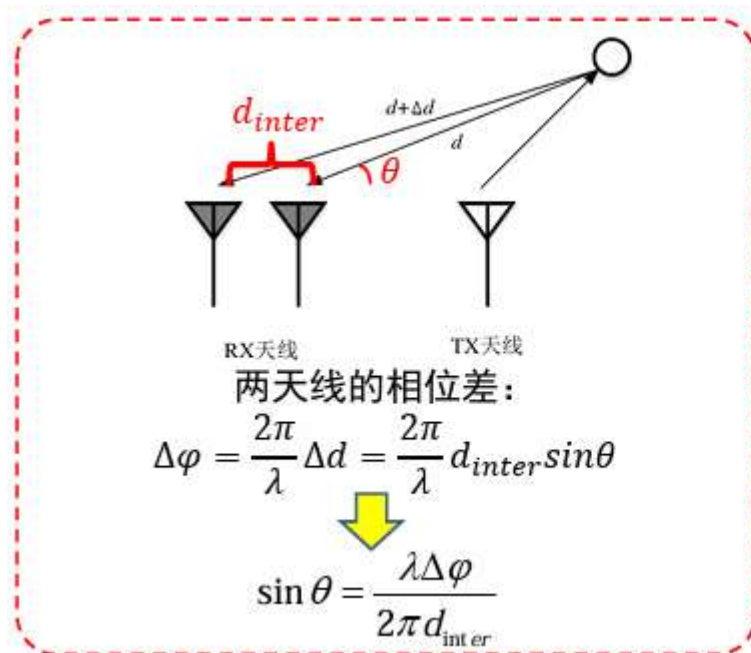
从目标到每个天线的差分距离 Δd 导致FFT峰值发生相位变化，该相位变化用于估计到达角 θ 。

手势识别信号处理

信号处理流程

□ 角度信息提取：MUSIC算法（详见附录2）

◆ 根据目标回波信号在不同天线间的相位变化，估算目标到不同天线的距离差

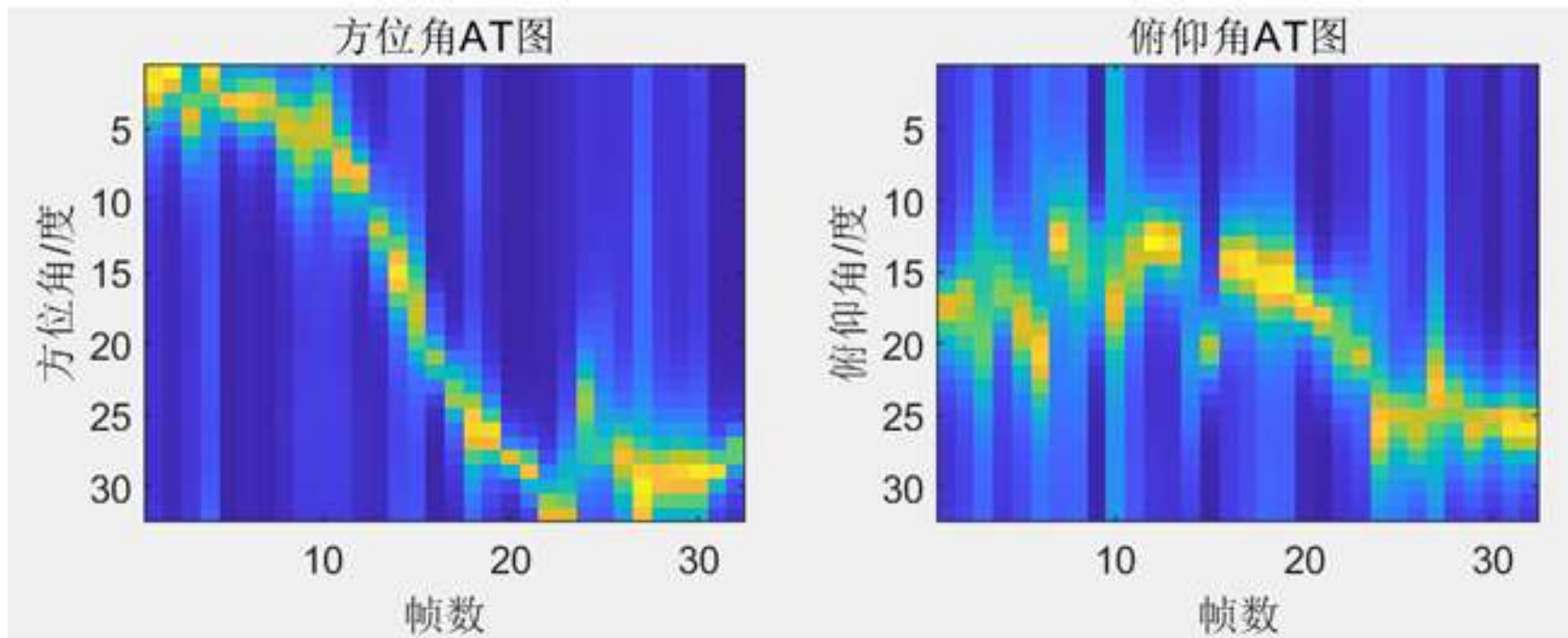


手势识别信号处理

信号处理流程

□ 角度信息表现形式：AT图（方位+俯仰）

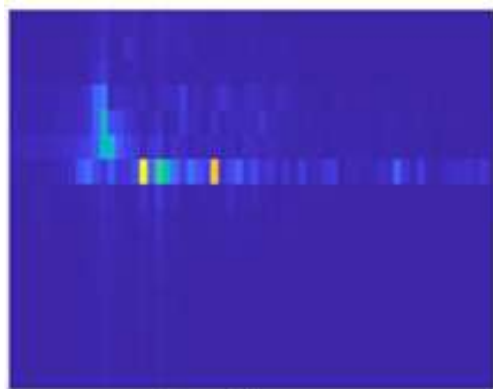
◆ 在两个方向上，角度随帧数变化



手势识别信号处理

信号处理流程

- 前面展示的RD图和AT图均为经过后处理的结果，原始图像无法直接使用
 - ◆ 静止目标成像干扰
 - ◆ 噪声干扰
 - ◆ 处理方式：MTI消除静止目标回波，恒虚警检测消除噪声影响



处理前RD图



处理后RD图

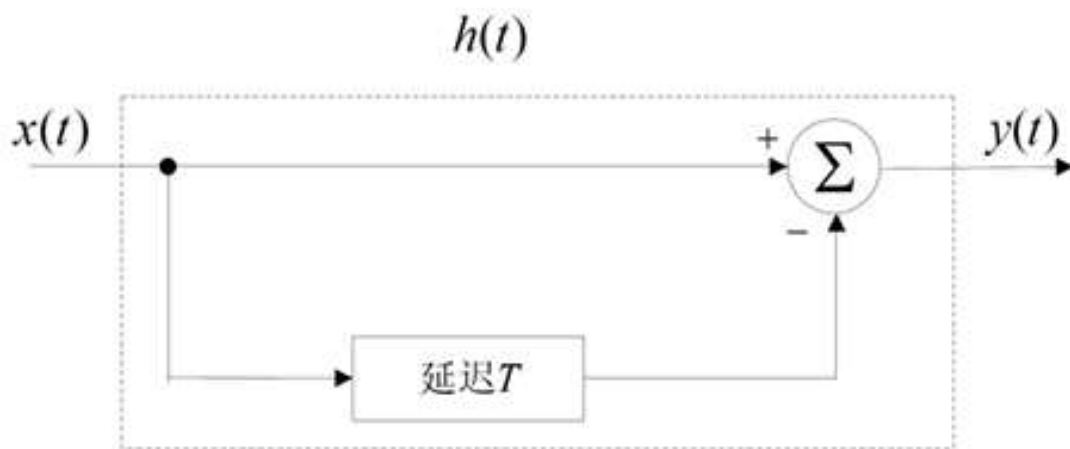
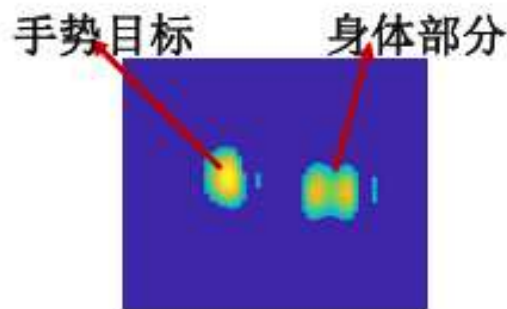
手势识别信号处理

信号处理流程

□ 两脉冲MTI对消

◆ 目的：消除雷达图像中的静止目标

- 运动目标的回波会随着脉冲的变化而变化，因此输出是非零的
- 静止杂波的回波是恒定的，因而受到抑制

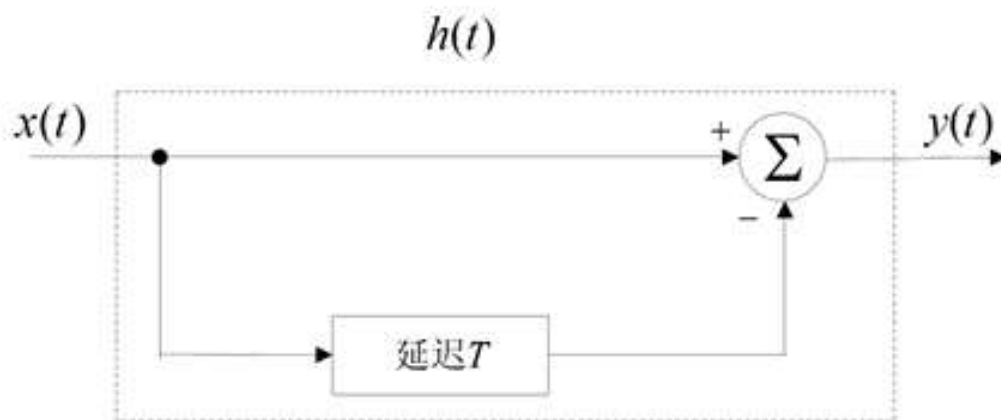


两脉冲MTI对消滤波器结构图

手势识别信号处理

信号处理流程

□ 两脉冲MTI对消



输出信号: $y(t) = x(t) - x(t - T)$

滤波器冲激响应: $h(t) = \delta(t) - \delta(t - T)$

频域表达式: $H(\omega) = 1 - e^{-j\omega T}$

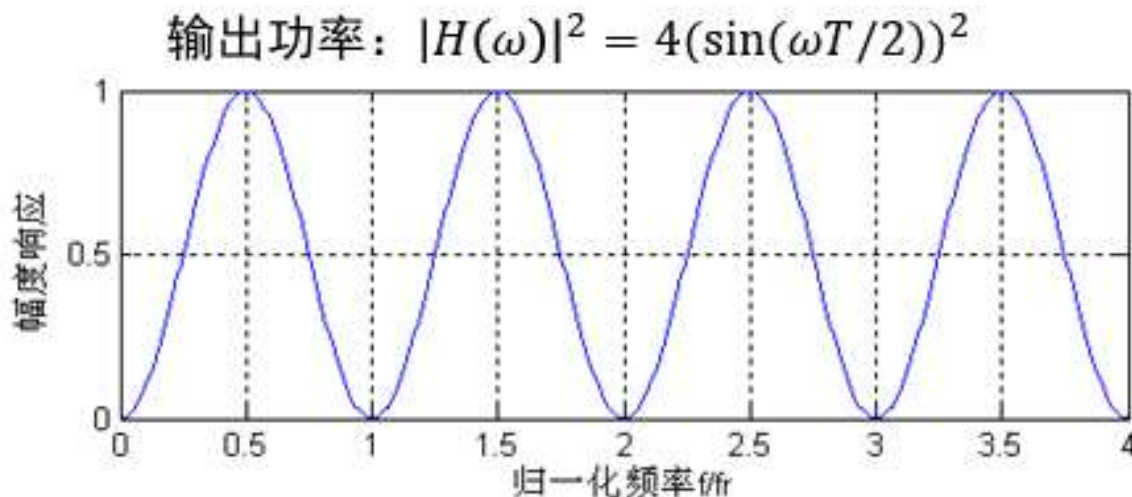
输出功率: $|H(\omega)|^2 = 4(\sin(\omega T/2))^2$

手势识别信号处理

信号处理流程

□ 两脉冲MTI对消

- ◆ 静止目标多普勒频率为0，经MTI脉冲对消滤波器被滤除
- ◆ 运动目标相邻脉冲回波相位差 $\Delta\varphi=2\pi f_d T_r$ ， $f_d=2vr/\lambda$
 - $\Delta\varphi=2\pi$ ，即 $f_d T_r=n$ 时， $vr=n\lambda/(2T_r)$ ，称为盲速
 - 设置发射信号参数中，应尝试间隔不同脉冲数量，找到对应手掌运动速度的最佳盲速

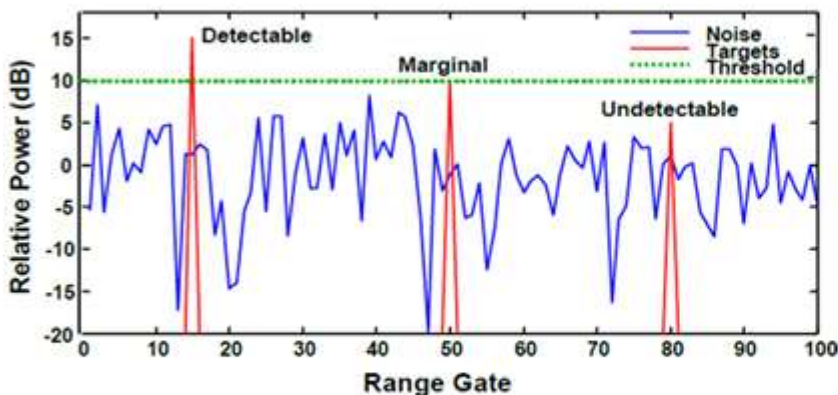


手势识别信号处理

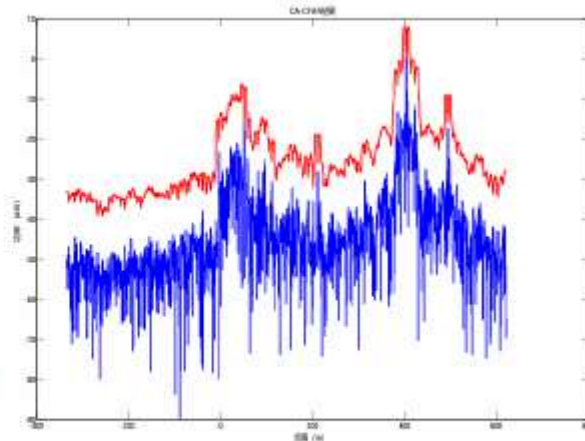
信号处理流程

恒虚警检测

- ◆ 在噪声环境下确定雷达量测值是目标回波还是干扰项
- ◆ 方法：划定门限，如果雷达回波大于门限，则为目标，否则为噪声



固定门限



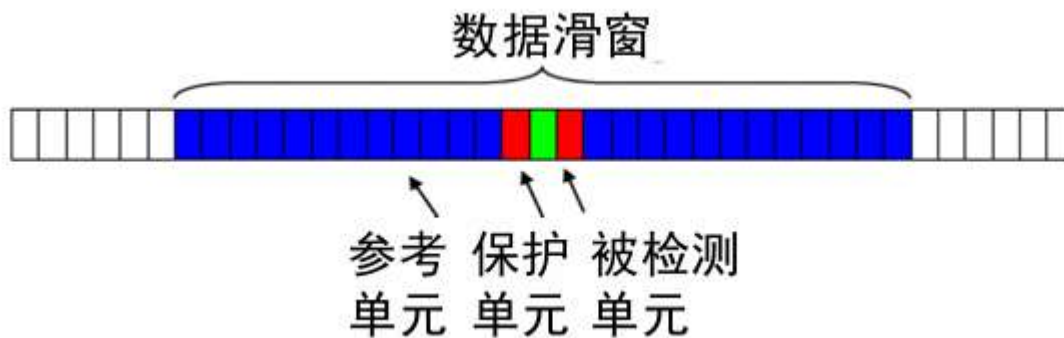
自适应门限

手势识别信号处理

信号处理流程

□ 恒虚警检测：CFAR处理

- ◆ 确定保护单元，参考单元和被检测单元
- ◆ 编写for循环来将数据滑窗遍历所有的数据（快时间和慢时间）
 - 函数：for循环，if判断

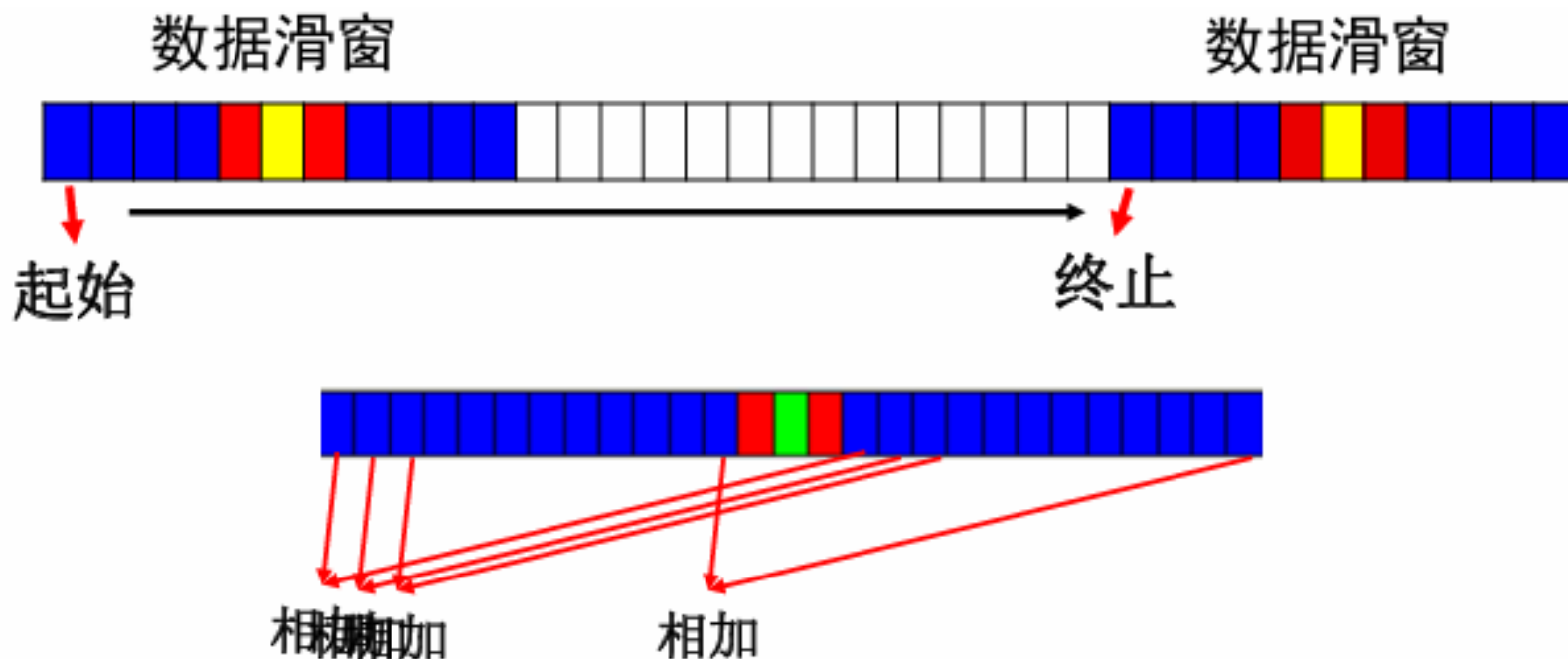


手势识别信号处理

信号处理流程

恒虚警检测：CFAR处理

- ◆ 针对快时间域的CFAR处理: for循环range范围=矩阵长度-窗长
- ◆ 对参考单元进行累加取平均

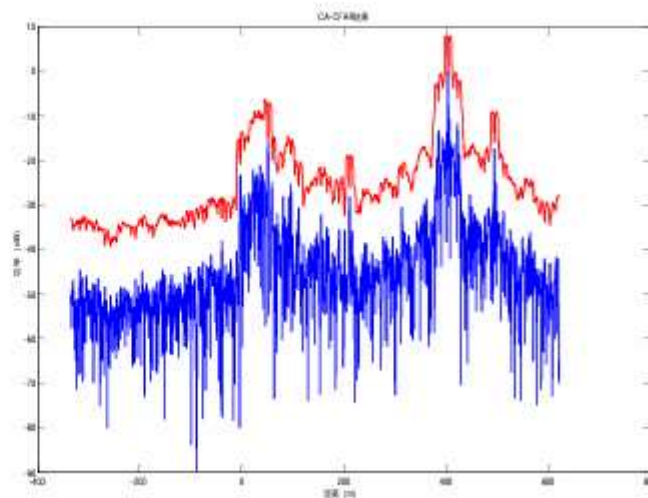
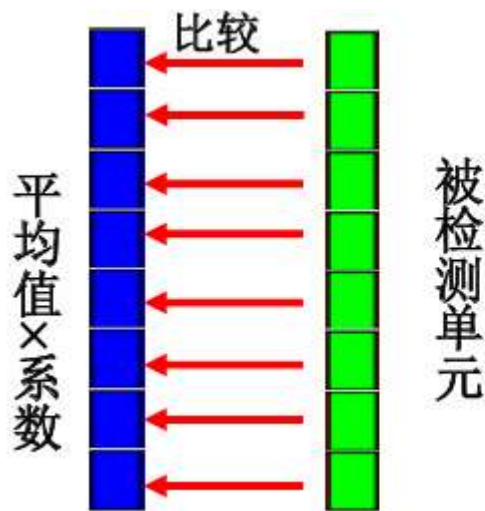


手势识别信号处理

信号处理流程

❑ 恒虚警检测：CFAR处理

- ◆ 针对慢时间域的CFAR处理：for循环range范围= 慢时间长度
- ◆ If 判断（自适应门限）
 - 被检测单元是否大于参考单元累加结果平均值 \times 系数
 - 系数取值需要大于1（通常范围是2-5）



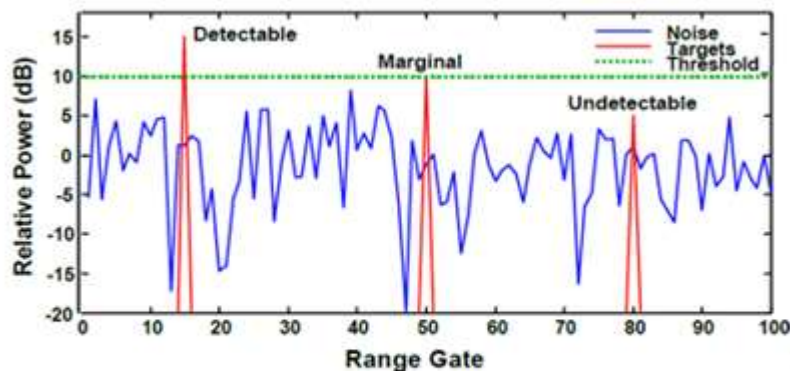
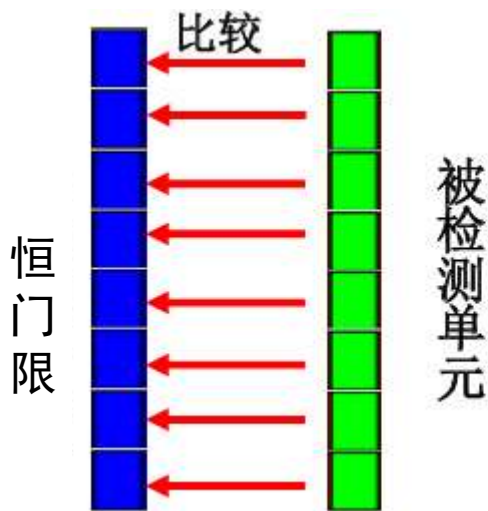
手势识别信号处理

信号处理流程

□ 恒虚警检测：固定门限处理

◆ If判断：

- 被检测单元是否大于恒门限（最大值 \times 系数），抑制低频杂波
- 系数取值需要小于1（通常范围是0.6-0.9）

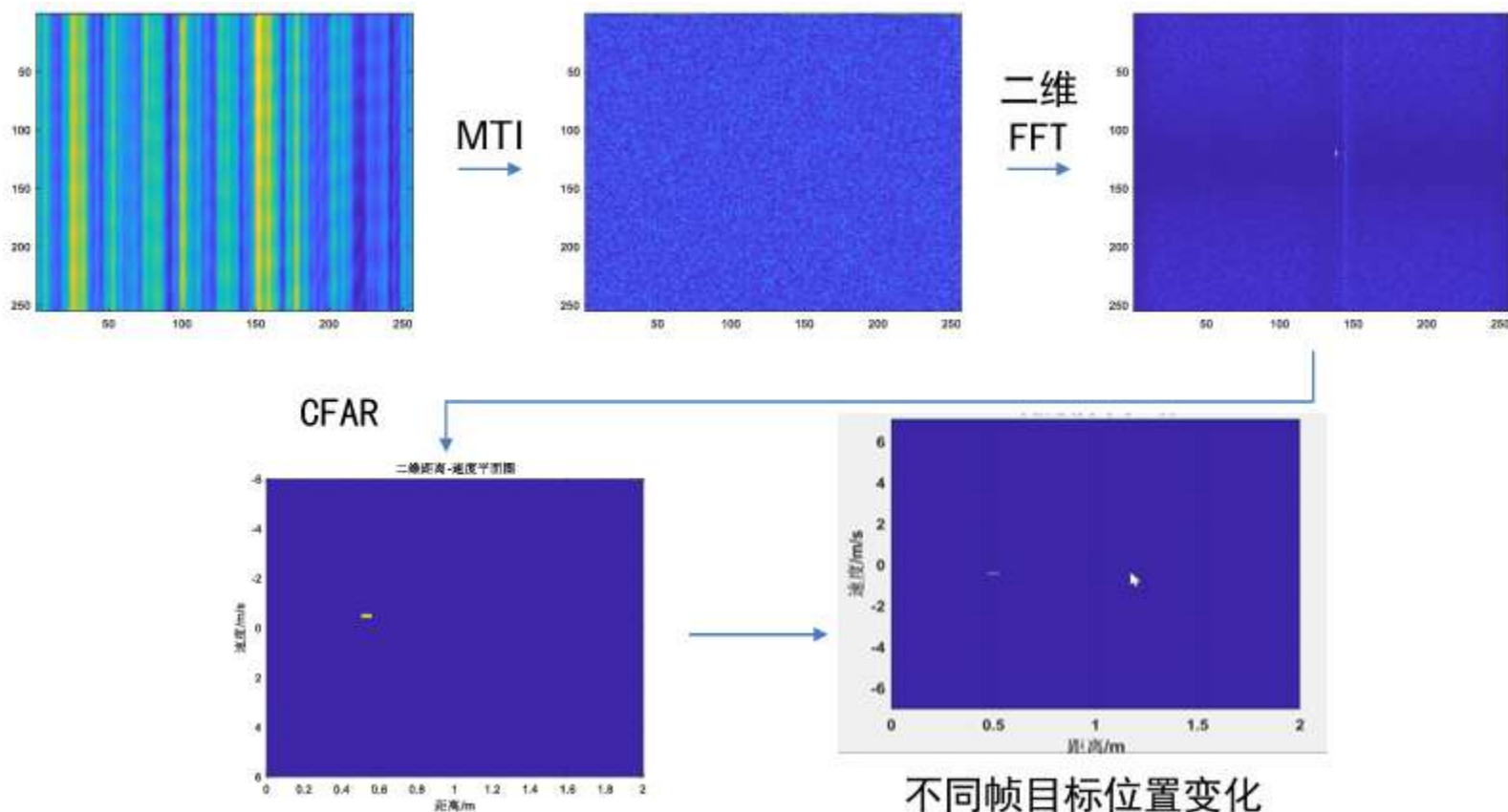


固定门限

手势识别信号处理

信号处理流程

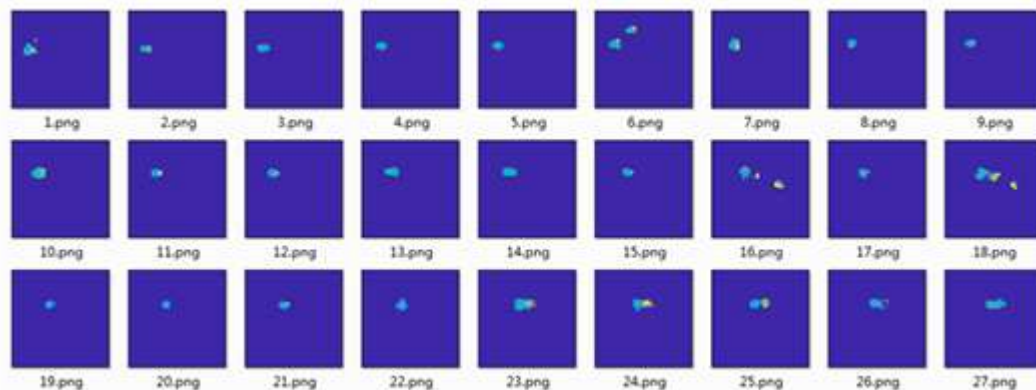
□ 处理流程实例



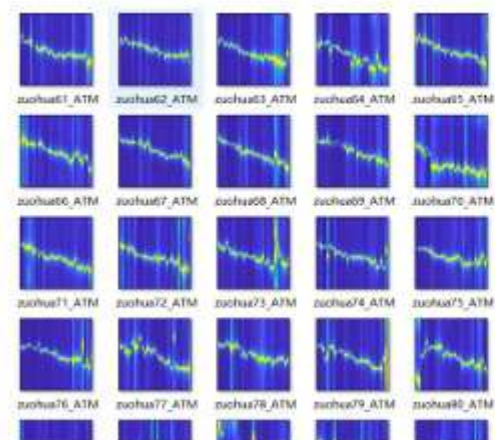
手势识别信号处理

信号处理流程

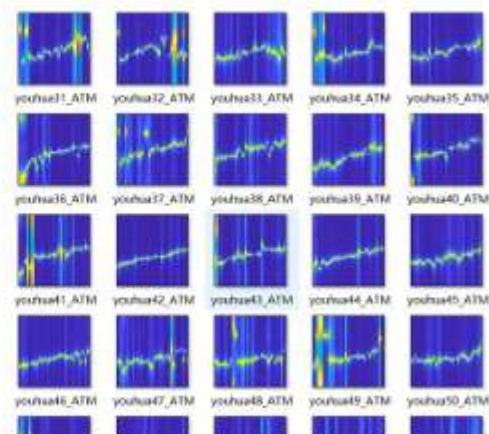
□ RD图序列



□ ATM



左划



右划

网络识别

网络设计流程

□ 融合神经网络

- ◆ 设计CNN网络对距离速度图序列进行特征提取
- ◆ 设计CNN网络对角度时间图进行特征提取
- ◆ 将上述两种特征进行融合，添加全连接层和输出层进行分类

□ 建议

- ◆ 数据扩充增加数据量（随机裁剪、随机旋转、随机高斯噪声）

网络识别

网络设计流程

□ 推荐使用conda环境配置，推荐使用pycharmIDE



Version: 2022.1.4
Build: 221.6008.17
21 July 2022

[System requirements](#)
[Installation instructions](#)
[Other versions](#)
[Third-party software](#)

Download PyCharm

[Windows](#) [macOS](#) [Linux](#)

Professional

For both Scientific and Web Python development. With HTML, JS, and SQL support.

[Download](#)

Free 30-day trial available

Community

For pure Python development

[Download](#)

Free, built on open-source



Get the Toolbox App to download PyCharm and its future updates with ease

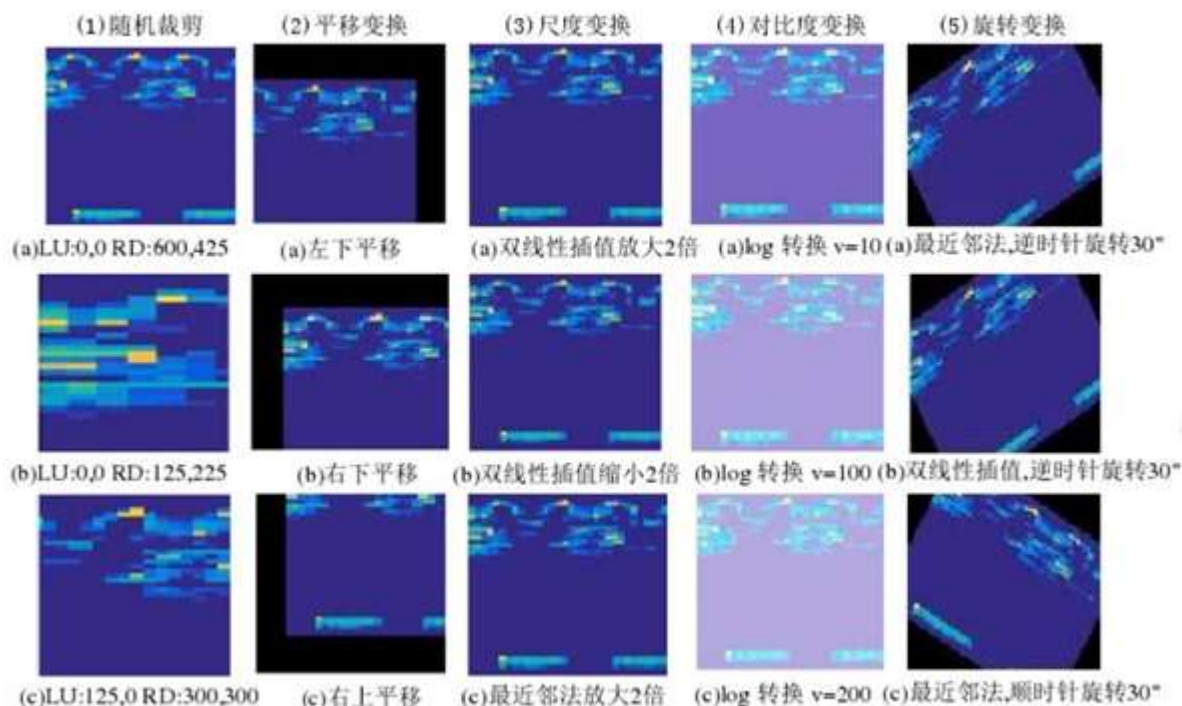
下载免费社区版

网络识别

网络设计流程

□ 数据扩充

- ◆ 通过加噪声、裁切、拉伸等操作，可以实现对训练集、测试集的数据扩充

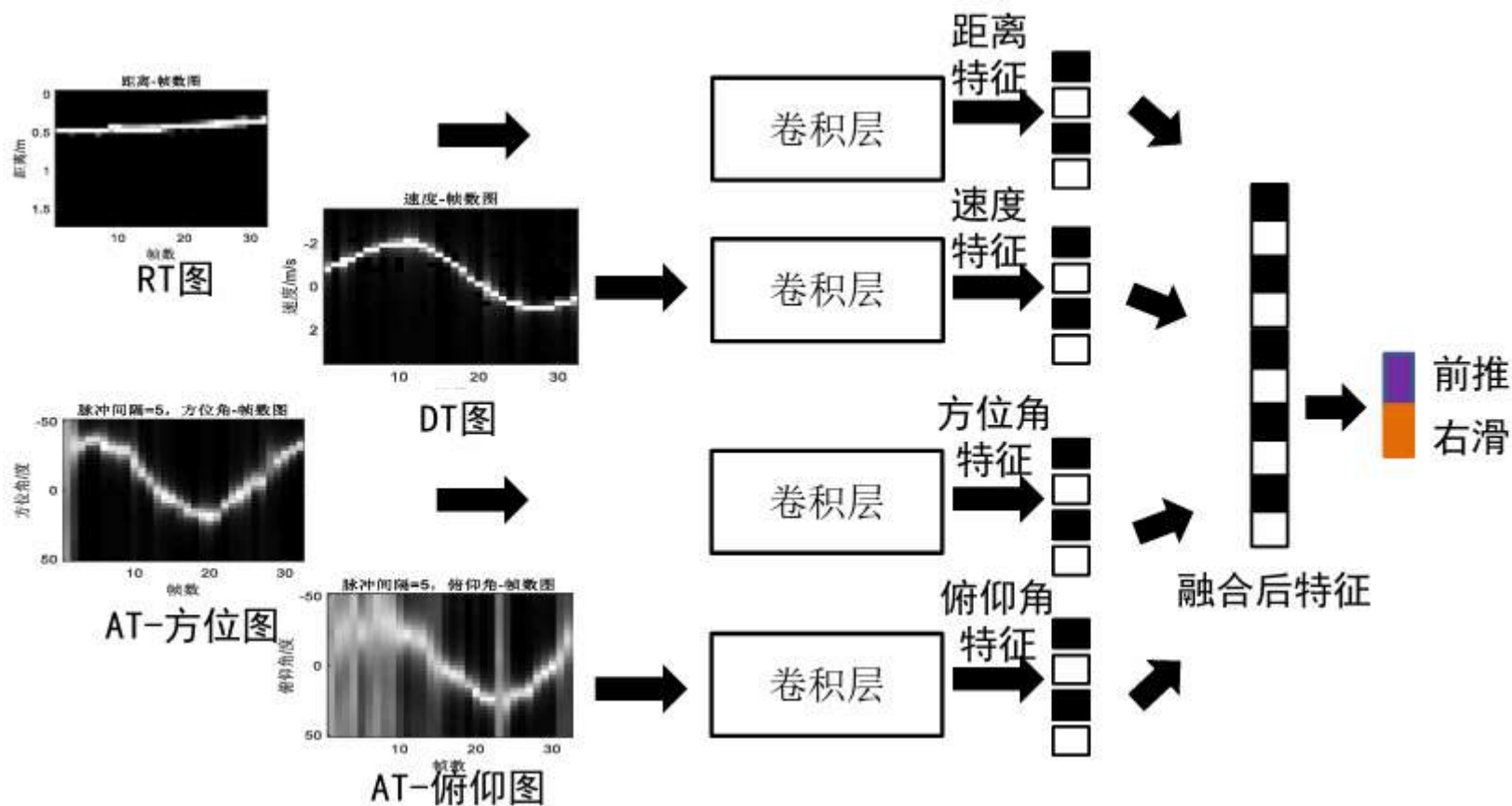


网络识别

网络设计流程

□ 特征融合：手势的距离、速度、角度（方位/俯仰）特征

◆ 先提取特征再融合

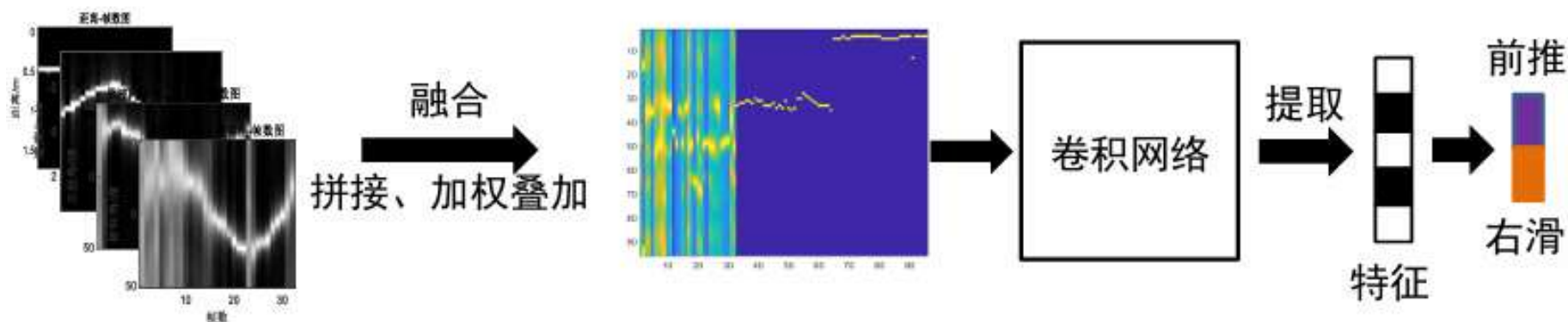


网络识别

网络设计流程

□ 特征融合：手势的距离、速度、角度（方位/俯仰）特征

◆ 先融合再提取特征

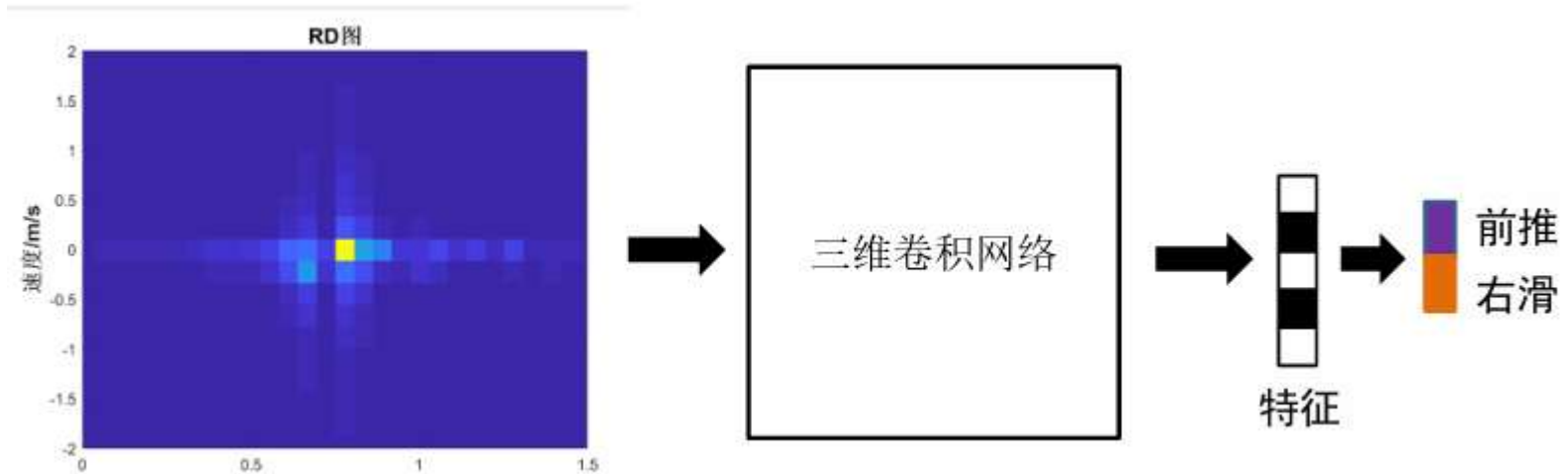


思路拓展

网络设计

□ 三维卷积神经网络

◆ 对RD图序列采用三维卷积神经网络



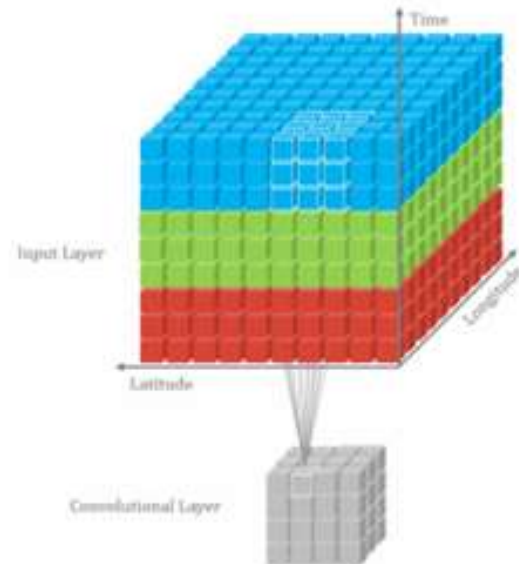
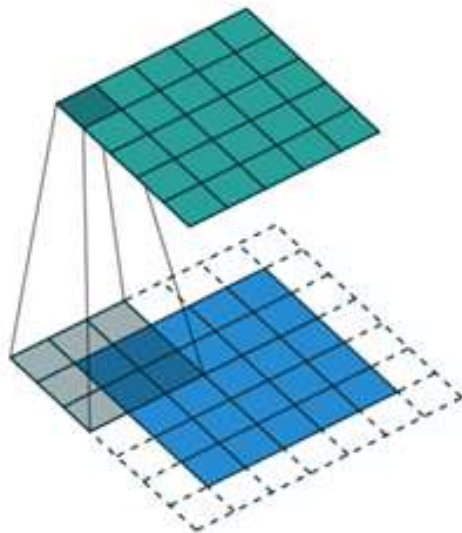
思路拓展

网络设计

□ 三维卷积神经网络

◆ 对RD图序列采用三维卷积神经网络

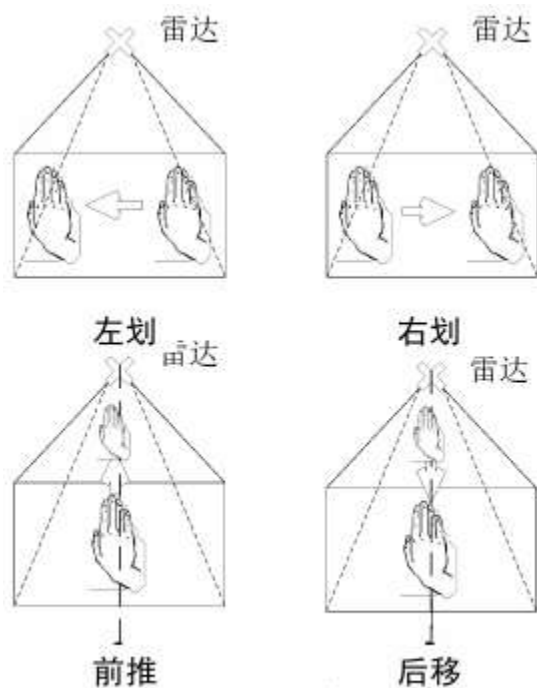
- 3D-CNN主要运用在视频分类、动作识别等领域，是在2D-CNN的基础上改变而来，可以捕获各帧之间的关联信息，能够提取到更具表达性的特征



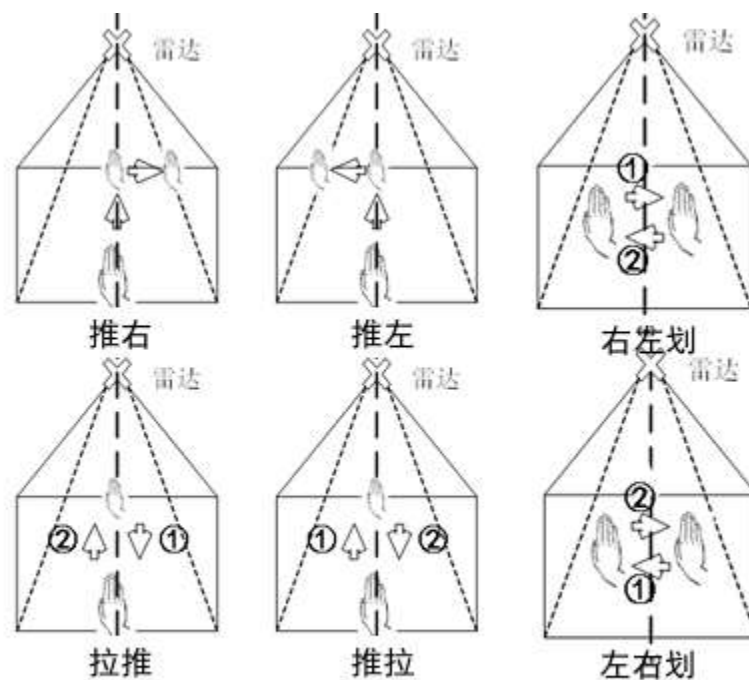
思路拓展

网络设计

- ❑ 进阶1：自行采集数据集，扩充手势数量至10种进行分类
- ❑ 进阶2：自行设计信号处理流程及网络结构，完成分类任务



基础



进阶

其他

◆ 建议

- 数据扩充（随机裁剪、随机旋转、随机高斯噪声）
- Batch-size参数
- 每一类手势重复250次（200个用于训练，50个用于测试）
- 读入数据接口，参考torch.utils.data中Dataset使用

```
class yourName(Dataset):
```

```
    def __init__(self, split='train'):
```

```
        def __len__(self):
```

```
            def __getitem__(self, index):
```

```
train_set = yourName(split='train')
```

```
val_set = yourName(split='val')
```

```
train_loader = DataLoader(train_set, batch_size=4, shuffle=True)
```

```
val_loader = DataLoader(val_set, batch_size=4, shuffle=False)
```

作业

- ◆ 作业题目：利用ResNet-101模型实现MNIST分类的训练和测试
- ◆ 作业要求：独立完成，可以网上搜索相关代码
- ◆ 提交格式
 - 完整的代码（附加文档说明代码如何运行）
 - 文档（记录实验结果，比如训练过程的损失函数变化、测试精度等）

谢谢！