

# 3.1. Group Tracking

## 3.1.高级算法设计

随着检测精度的提高，真实世界的雷达目标（汽车、行人、墙壁、着陆地面等）。作为一组多个反射点的形式呈现给跟踪处理层。这些探测点形成了一组与范围、角度和角速度相关的测量值。当然，在任何时候都可能有多组现实世界的目标。因此，我们寻找一种能够与多个目标群体一起工作的跟踪器。组跟踪方法如下图所示。

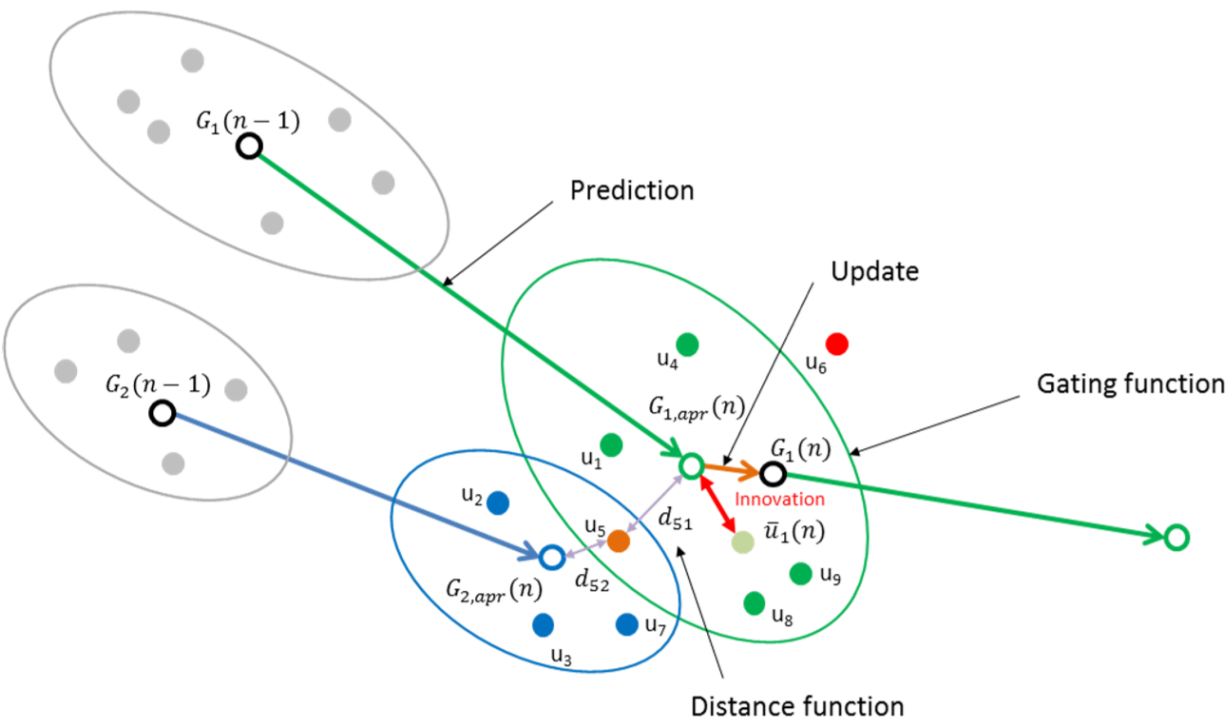


Figure 5. Group Tracking

<https://blog.csdn.net/djfkj52>

## 3.2. 群组跟踪框图

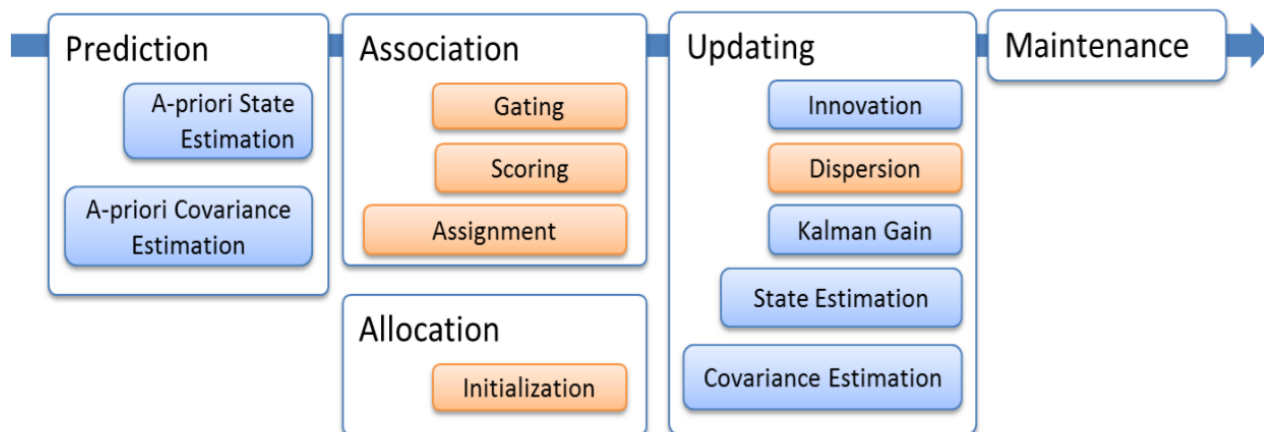


Figure 6. Tracking Block Diagram

<https://blog.csdn.net/djfkj52>

### 3.3. Prediction Step

我们使用卡尔曼滤波器预测过程，根据在时间 $n-1$ 估计的状态和过程协方差矩阵，估计时间 $n$ 的跟踪组质心。我们为每个可跟踪对象计算先验状态和协方差估计。在这一步，我们还计算测量矢量估计。

### 3.4. Association Step

假设存在一个或多个tracks以及相关的预测状态向量。对于每个给定的track，我们形成一个关于预测质心的门gate。

gate口应考虑以下因素：

目标操纵maneuver 组的分散dispersion of the group 测量噪声

我们使用组残差协方差矩阵在跟踪组质心周围的3D测量空间中构建椭球。椭球将代表gating函数，以限定我们在时间 $n$ 观察到的单个测量结果。gating设计在以下部分中说明

对于gate的测量，我们将归一化距离函数作为cost函数进行计算，以将测量与每个轨迹相关联

assignment process将cost函数最小化，一次将一项测量分配给最近的track。这将创建与每个轨道相关的一组测量

...3.5 3.6

# 4.Implementation Details

## 4.1. Group Tracker

跟踪算法被实现为一个库。应用程序任务使用配置参数创建算法实例，该配置参数描述传感器，场景和雷达目标的行为。从应用程序任务上下文每帧调用一次算法。可以创建组跟踪器的多个实例。

下图说明了算法在每个帧调用期间执行的步骤。算法以极坐标（范围，角度，多普勒）输入测量数据，并在笛卡尔空间中跟踪对象。因此，我们使用扩展卡尔曼滤波器（EKF）流程。

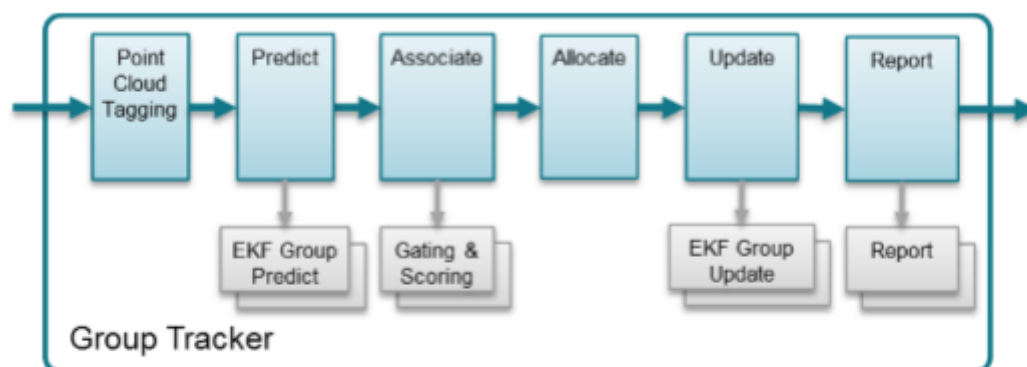


Figure 9. Group Tracking Algorithm <https://blog.csdn.net/djfkj52>

首先基于场景边界标记点云输入。有些点可能被标记为“边界外”，在关联和分配过程中将被忽略。

预测函数根据在时间 $n-1$ 估计的状态和过程协方差矩阵估计时间 $n$ 的跟踪组质心。我们为每个可跟踪对象计算先验状态和误差协方差估计。在这一步，我们还计算测量矢量估计。

关联功能允许每个跟踪单元指示每个测量点是否“足够近”（门控），如果是，则提供出价值（scoring）。point分配给最高出价者。未分配的点正在通过分配功能。

在分配过程中，首先根据点在测量坐标中的接近度将它们合并为一组集合。每个集合成为分配决策的候选者。它必须通过多项测试才能成为新的track。一旦通过，就分配新的跟踪单元。

在“更新”步骤中，将根据一组关联点来更新track。我们计算innovation，卡尔曼增益以及后验状态向量和误差协方差。

除经典EKF外，误差协方差计算还包括测量噪声协方差矩阵中的 group dispersion。

报告功能查询每个跟踪单元并产生算法输出。

## 4.2. Building and using the library

该算法以现成的makefile基础结构作为源代码提供。可以构建库来支持2D或3D几何。应用程序应包括库（2D或3D），因为它需要。以流量监控（TM）和人数统计（PC）用例的建议值为例。

与以前（仅2D）版本的更改以红色突出显示。配置将在模块创建时传递给算法实例。下一章介绍了配置参数。

## 4.3. Configuration Parameters

配置参数用于配置跟踪算法。它们应根据特定的场景和目标特性进行调整，以匹配客户用例。参数分为强制性和可选（高级）。强制性参数如下所述。

### 4.3.1.强制性配置参数

参数	T M	P C	单 位	描述
maxNumPoi nts	2 5 0	250	-	每一帧的最大检测点云数
maxNumTra cks	2 0	20	-	在任何给定时间内要跟踪的最大目标数量
stateTrackin gVectorTyp e	2 D A	2D A	-	2DA={x,y,vx,vy,ax,ay} 3DA={x,y,z,vx,vy,vz,ax,ay,az}
initialRadial Velocity	- 2 0	0	m/ s	在探测时刻的预期目标径向速度
maxRadialV elocity	N/ A	N/A	m/ s	由传感器报告的最大绝对径向速度。这应与传感器的ch irp配置相匹配
radialVelocit yResolution	N/ A	N/A	m/ s	由传感器报告的最小非零径向速度。这应与传感器的ch irp配置相匹配
maxAcceler ation	0, 2 0, 0	0.1, 0.1, 0.1	m/ s ^ 2	横向、纵向和垂直方向的最大目标加速度。用于计算处 理噪声矩阵。对于2D选项，将忽略垂直方向
deltaT	N/ A	N/A	m s	帧速率。这应与传感器的chirp配置相匹配

参数	T M	PC	单 位	描述
verbosityLevel	N O N E	NO NE	-	表示冗长级别的位掩码：NONE   WARNING

### 4.3.2. Advanced parameters

高级参数分为几组。每个集合都可以省略，默认值将由算法使用。客户应修改所需的参数以获得更好的性能。

#### Scenery Parameters

这组参数描述了场景。它允许用户为跟踪器配置预期的边界和静态行为区域。用户最多可以定义2个边界框和2个静态框。方框坐标以米为单位，传感器位于笛卡尔 (X, Y) 空间的 (0, 0, 0) 。

Table 2. Scenery Parameters

Parameter	TM	PC		Description
numBoundaryBoxes	1U	1U	-	Number of boundary boxes defined. Points outside boundary box will be ignored
boundaryBox[2]	{-1.f,12.f, 15.f,75.f,0.f,0.f}, {0.f,0.f,0.f,0.f,0.f,0.f}	{-4.f,4.f, 0.5.f,7.5f,0.f,0.f}, {0.f,0.f,0.f,0.f,0.f,0.f}	m	{x1,x2, y1,y2,z1,z2 }
numStaticBoxes	1U	1U	-	Number of static boxes defined. Targets inside static box are allowed to persist as static
staticBox[2]	{0.f, 11.f,19.f,50.f,0.f,0.f}, {0.f,0.f,0.f,0.f,0.f,0.f}	{-3.f,3.f,2.f,6.f,0.f,0.f}, {0.f,0.f,0.f,0.f,0.f,0.f}	m	{x1,x2, y1,y2,z1,z2 }

<https://blog.csdn.net/djfkj52>

numBoundaryBoxes：已定义的边界框数。边界框之外的点将被忽略

numStaticBoxes：已定义的静态方框数。静态框内的目标被允许作为静态框持续存在

#### Measurement Standard Deviation Parameters 测量标准偏差参数

这一组参数被忽略了

**Allocation Parameters 分配参数**