Table of Contents


```
% main_cumulative_cfar_analysis_v2.m
%
% □是一个使用并行□算□行□化的累□ CFAR □果分析主控脚本。
% 1. 将主循□从 for 更改□ parfor，以利用多核 CPU 并行□理，□著提高运行速度。
% 2. □整了数据收集□□以适□ parfor 的工作机制。
%
% 修改□□
% date       by      version                  modify
% 25/07/07   XZR      v1.0      在 main_cumulative_cfar_analysis.m 的基□上引入
parfor 加速信号□理
%
% 未来改□：
clc; clear; close all;
```

1. 用□配置区

--- □□□□ --- 'ppi': □制累□的平面位置□示□ (极坐□) 'rdm': □制累□的距离-多普勒□ (矩形坐□) 'az_range': □制累□的距离-方位□ (矩形坐□) 'summary': □制包含以上所有信息的□合□表□

```
plot_option = 'summary'; % <--- 在此□□最□的□□□型
frame_range = 0:151;
% --- 保存功能开关 ---
options.save_detection_log = true;
options.save_cumulative_image = true;
% --- 伺服角度修正 ---
NorthAngle = 307;
FixAngle = 35;
```

2. 路径与参数配置

```
n_exp = 3;
win_size = 4;
T_CFAR = 7;
base_path  = uigetdir('', '□□□数据根目□');
if isequal(base_path, 0), disp('用□取消了文件□□。'); return; end
mtd_data_path = fullfile(base_path, num2str(n_exp), ['MTD_data_win',
num2str(win_size)]);
cfar_data_path = fullfile(base_path, num2str(n_exp), ['cfarFlag4_T',
```

```
num2str(T_CFAR)]);
header_data_path = fullfile(base_path, num2str(n_exp),
'Framehead_information');
output_path = fullfile(base_path, num2str(n_exp), 'Cumulative_Results');
if ~exist(output_path, 'dir'), mkdir(output_path); end

% --- 雷达系□参数 ---
params.c = 2.99792458e8;
params.prtNum = 332;
params.prt = 232.76e-6;
params.fs = 25e6;
params.fc = 9450e6;
params.beam_num = 13;
params.point_prt_total = 3404;
params.prf = 1 / params.prt;
params.wavelength = params.c / params.fc;
params.deltaR = params.c / (2 * params.fs);

% --- 速度范□□置参数 ---
velocity.lowerSpeedBound = -2;
velocity.upperSpeedBound = 2;
```

3. 初始化与□□算

--- 在所有循□开始前，□先□算好速度□ ---

```
v_axis = linspace(-params.prf/2, params.prf/2, params.prtNum) *
params.wavelength / 2;
```

4. 并行□理主循□

```
fprintf('--- 开始并行□理，提取所有 CFAR □□点 ---\n');
fprintf('正在启□并行池...\n');
tic; % 开始□□

% --- □置□度条 ---
try
    p = gcp; % □取当前的并行池
catch
    % 如果没有并行池，□□建一个
    p = parpool;
end

N = length(frame_range); % □取□的迭代次数
D = parallel.pool.DataQueue; % □建一个数据□列
h = waitbar(0, '正在初始化并行任□...'); % □建一个□度条窗□

% 将□数器存□在 waitbar 的 UserData 属性中
h.UserData = 0;

% 使用匿名函数将 waitbar 句柄和□数 N □□□回□函数
afterEach(D, @(~) updateWaitbar(h, N));
```

```
% --- 【核心□化】将 for 更改□ parfor ---
% parfor 会将 frame_range 中的任□分配□多个 worker 并行□行
% □建一个元胞数□来收集每个 worker 的返回□果
% --- parfor 循□ ---
parfor_results = cell(1, N);
parfor i = 1 : N % 在 parfor 循□中，避免使用 fprintf 来□示□度，□出会混乱。
    frame_idx = frame_range(i);

    % --- 每个 worker 独立加□所需文件 ---
    mtd_file = fullfile(mtd_data_path, ['frame_', num2str(frame_idx),
'.mat']);
    cfar_file = fullfile(cfar_data_path, ['frame_', num2str(frame_idx),
'.mat']);
    header_file = fullfile(header_data_path, ['frame_', num2str(frame_idx),
'.mat']);

    if ~exist(mtd_file, 'file') || ~exist(cfar_file, 'file') ||
~exist(header_file, 'file')
        warning('□ #%d 的文件缺失，跳□此□。', frame_idx);
        continue; % 跳到下一次 parfor 迭代
    end

    % 使用 load 函数加□数据到□□□构体中
    mtd_data = load(mtd_file, 'MTD_win_all_beams');             % mtd 好像不怎么
用的上
    cfar_data = load(cfar_file, 'cfarFlag_win_all_beams');
    header_data = load(header_file, 'FrameHead_information');  % □个最好能用子
函数直接□取

    % --- 初始化当前 worker 的局部日志□量 ---
    % 每个 worker 都□□一个自己的日志，最后再合并，□是 parfor 的□准□践
    local_detection_log = [];

    for b = 1:params.beam_num
        for s = 1:win_size
            mtd_slice = squeeze(mtd_data.MTD_win_all_beams{b}(s, :, :));
            cfar_flag = squeeze(cfar_data.cfarFlag_win_all_beams{b}
(s, :, :));

            [detected_v_indices, detected_r_indices] = find(cfar_flag);

            if ~isempty(detected_v_indices)
                % --- 向量化□算 ---
                num_detections = numel(detected_v_indices);
                % prt_info = header_data.FrameHead_information(1);
                prt_info = header_data.FrameHead_information(1);
                velocities = v_axis(detected_v_indices).';
                ranges = detected_r_indices * params.deltaR;
                linear_indices = sub2ind(size(mtd_slice),
detected_v_indices, detected_r_indices);
                snrs = mtd_slice(linear_indices);

                % 方位角（伺服角）修正，画□函数已□修正了
                current_servo_angle = prt_info.current_servo_angle;
```

```
                % current_servo_angle_corrected =
fun_correct_servo_angle(current_servo_angle, NorthAngle, FixAngle);  % 口用伺服
角修正子函数修正伺服角

                % --- 批量口建口构体 ---
                frame_cell = num2cell(repmat(frame_idx, num_detections, 1));
                beam_cell = num2cell(repmat(b, num_detections, 1));
                slice_cell = num2cell(repmat(s, num_detections, 1));
                prt_index_cell = num2cell(repmat(prt_info.pulse_no,
num_detections, 1));
                range_bin_cell = num2cell(detected_r_indices);

                azimuth_cell = num2cell(repmat(current_servo_angle,
num_detections, 1)); % 方位角口元

                elevation_cell = num2cell(nan(num_detections, 1)); % 俯仰角口元
                velocity_cell = num2cell(velocities);
                range_m_cell = num2cell(ranges);
                snr_cell = num2cell(snrs);
                timestamp_cell = num2cell(repmat(prt_info.timer_cnt,
num_detections, 1));

                current_detections = struct(...
                    'frame', frame_cell, 'beam', beam_cell, 'slice',
slice_cell, ...
                    'prt_index', prt_index_cell, 'range_bin',
range_bin_cell, ...
                    'azimuth_deg', azimuth_cell, 'elevation_deg',
elevation_cell, ...
                    'velocity_ms', velocity_cell, 'range_m', range_m_cell,
...
                    'snr', snr_cell, 'timestamp', timestamp_cell);

                % 将当前找到的目口点追加到局部的日志中
                local_detection_log = [local_detection_log;
current_detections];
            end
        end
    end

    % 将当前 worker 口理完一整口的口果，存入口的口果元胞数口中
    parfor_results{i} = local_detection_log;

    % --- 在每次迭代口束口，向口列口送一个信号 ---
    send(D, i);
end
toc; % 口束口口并口示口耗口
```

```
--- 开始并行口理，提取所有 CFAR 口口点 ---
正在启口并行池...
警告：口 #3 的文件缺失，跳口此口。
警告：口 #34 的文件缺失，跳口此口。
警告：口 #33 的文件缺失，跳口此口。
警告：口 #32 的文件缺失，跳口此口。
```

警告：□ #31 的文件缺失，跳□此□。
警告：□ #30 的文件缺失，跳□此□。
警告：□ #29 的文件缺失，跳□此□。
警告：□ #28 的文件缺失，跳□此□。
警告：□ #27 的文件缺失，跳□此□。
警告：□ #26 的文件缺失，跳□此□。
警告：□ #133 的文件缺失，跳□此□。
警告：□ #132 的文件缺失，跳□此□。
警告：□ #149 的文件缺失，跳□此□。
警告：□ #0 的文件缺失，跳□此□。
警告：□ #16 的文件缺失，跳□此□。
警告：□ #15 的文件缺失，跳□此□。
警告：□ #14 的文件缺失，跳□此□。
警告：□ #13 的文件缺失，跳□此□。
警告：□ #12 的文件缺失，跳□此□。
警告：□ #11 的文件缺失，跳□此□。
警告：□ #10 的文件缺失，跳□此□。
警告：□ #9 的文件缺失，跳□此□。
警告：□ #8 的文件缺失，跳□此□。
警告：□ #89 的文件缺失，跳□此□。
警告：□ #88 的文件缺失，跳□此□。
警告：□ #87 的文件缺失，跳□此□。
警告：□ #86 的文件缺失，跳□此□。
警告：□ #85 的文件缺失，跳□此□。
警告：□ #114 的文件缺失，跳□此□。
警告：□ #113 的文件缺失，跳□此□。
警告：□ #112 的文件缺失，跳□此□。
警告：□ #111 的文件缺失，跳□此□。
警告：□ #110 的文件缺失，跳□此□。
警告：□ #125 的文件缺失，跳□此□。
警告：□ #124 的文件缺失，跳□此□。
警告：□ #136 的文件缺失，跳□此□。
警告：□ #142 的文件缺失，跳□此□。
警告：□ #146 的文件缺失，跳□此□。
警告：□ #2 的文件缺失，跳□此□。
警告：□ #52 的文件缺失，跳□此□。
警告：□ #51 的文件缺失，跳□此□。
警告：□ #50 的文件缺失，跳□此□。
警告：□ #49 的文件缺失，跳□此□。
警告：□ #48 的文件缺失，跳□此□。
警告：□ #47 的文件缺失，跳□此□。
警告：□ #46 的文件缺失，跳□此□。
警告：□ #45 的文件缺失，跳□此□。
警告：□ #44 的文件缺失，跳□此□。
警告：□ #127 的文件缺失，跳□此□。
警告：□ #126 的文件缺失，跳□此□。
警告：□ #139 的文件缺失，跳□此□。
警告：□ #147 的文件缺失，跳□此□。
警告：□ #6 的文件缺失，跳□此□。
警告：□ #61 的文件缺失，跳□此□。
警告：□ #60 的文件缺失，跳□此□。
警告：□ #59 的文件缺失，跳□此□。
警告：□ #58 的文件缺失，跳□此□。
警告：□ #57 的文件缺失，跳□此□。

警告：□ #56 的文件缺失，跳□此□。
警告：□ #55 的文件缺失，跳□此□。
警告：□ #54 的文件缺失，跳□此□。
警告：□ #53 的文件缺失，跳□此□。
警告：□ #94 的文件缺失，跳□此□。
警告：□ #93 的文件缺失，跳□此□。
警告：□ #92 的文件缺失，跳□此□。
警告：□ #91 的文件缺失，跳□此□。
警告：□ #90 的文件缺失，跳□此□。
警告：□ #119 的文件缺失，跳□此□。
警告：□ #118 的文件缺失，跳□此□。
警告：□ #117 的文件缺失，跳□此□。
警告：□ #116 的文件缺失，跳□此□。
警告：□ #115 的文件缺失，跳□此□。
警告：□ #129 的文件缺失，跳□此□。
警告：□ #128 的文件缺失，跳□此□。
警告：□ #137 的文件缺失，跳□此□。
警告：□ #143 的文件缺失，跳□此□。
警告：□ #148 的文件缺失，跳□此□。
警告：□ #151 的文件缺失，跳□此□。
警告：□ #5 的文件缺失，跳□此□。
警告：□ #79 的文件缺失，跳□此□。
警告：□ #78 的文件缺失，跳□此□。
警告：□ #77 的文件缺失，跳□此□。
警告：□ #76 的文件缺失，跳□此□。
警告：□ #75 的文件缺失，跳□此□。
警告：□ #74 的文件缺失，跳□此□。
警告：□ #73 的文件缺失，跳□此□。
警告：□ #72 的文件缺失，跳□此□。
警告：□ #71 的文件缺失，跳□此□。
警告：□ #4 的文件缺失，跳□此□。
警告：□ #43 的文件缺失，跳□此□。
警告：□ #42 的文件缺失，跳□此□。
警告：□ #41 的文件缺失，跳□此□。
警告：□ #40 的文件缺失，跳□此□。
警告：□ #39 的文件缺失，跳□此□。
警告：□ #38 的文件缺失，跳□此□。
警告：□ #37 的文件缺失，跳□此□。
警告：□ #36 的文件缺失，跳□此□。
警告：□ #35 的文件缺失，跳□此□。
警告：□ #84 的文件缺失，跳□此□。
警告：□ #83 的文件缺失，跳□此□。
警告：□ #82 的文件缺失，跳□此□。
警告：□ #81 的文件缺失，跳□此□。
警告：□ #80 的文件缺失，跳□此□。
警告：□ #104 的文件缺失，跳□此□。
警告：□ #103 的文件缺失，跳□此□。
警告：□ #102 的文件缺失，跳□此□。
警告：□ #101 的文件缺失，跳□此□。
警告：□ #100 的文件缺失，跳□此□。
警告：□ #109 的文件缺失，跳□此□。
警告：□ #108 的文件缺失，跳□此□。
警告：□ #107 的文件缺失，跳□此□。
警告：□ #106 的文件缺失，跳□此□。

```
警告: □ #105 的文件缺失，跳□此□。
警告: □ #123 的文件缺失，跳□此□。
警告: □ #122 的文件缺失，跳□此□。
警告: □ #138 的文件缺失，跳□此□。
警告: □ #141 的文件缺失，跳□此□。
警告: □ #145 的文件缺失，跳□此□。
警告: □ #150 的文件缺失，跳□此□。
警告: □ #7 的文件缺失，跳□此□。
警告: □ #70 的文件缺失，跳□此□。
警告: □ #69 的文件缺失，跳□此□。
警告: □ #68 的文件缺失，跳□此□。
警告: □ #67 的文件缺失，跳□此□。
警告: □ #66 的文件缺失，跳□此□。
警告: □ #65 的文件缺失，跳□此□。
警告: □ #64 的文件缺失，跳□此□。
警告: □ #63 的文件缺失，跳□此□。
警告: □ #62 的文件缺失，跳□此□。
警告: □ #1 的文件缺失，跳□此□。
警告: □ #25 的文件缺失，跳□此□。
警告: □ #24 的文件缺失，跳□此□。
警告: □ #23 的文件缺失，跳□此□。
警告: □ #22 的文件缺失，跳□此□。
警告: □ #21 的文件缺失，跳□此□。
警告: □ #20 的文件缺失，跳□此□。
警告: □ #19 的文件缺失，跳□此□。
警告: □ #18 的文件缺失，跳□此□。
警告: □ #17 的文件缺失，跳□此□。
警告: □ #99 的文件缺失，跳□此□。
警告: □ #98 的文件缺失，跳□此□。
警告: □ #97 的文件缺失，跳□此□。
警告: □ #96 的文件缺失，跳□此□。
警告: □ #95 的文件缺失，跳□此□。
警告: □ #121 的文件缺失，跳□此□。
警告: □ #120 的文件缺失，跳□此□。
警告: □ #131 的文件缺失，跳□此□。
警告: □ #130 的文件缺失，跳□此□。
警告: □ #135 的文件缺失，跳□此□。
警告: □ #134 的文件缺失，跳□此□。
警告: □ #140 的文件缺失，跳□此□。
警告: □ #144 的文件缺失，跳□此□。
□□ 1.481280 秒。
```

5. 合并所有 worker 的□果

```
fprintf('--- 正在合并所有并行任□的□果 ---\n');
close(h); % 关□□度条
% 使用 vertcat 和 cellfun 将所有非空的局部日志合并成一个□的日志
detection_log = vertcat(parfor_results{:});
```

--- 正在合并所有并行任□的□果 ---

6. □示、保存最□□果

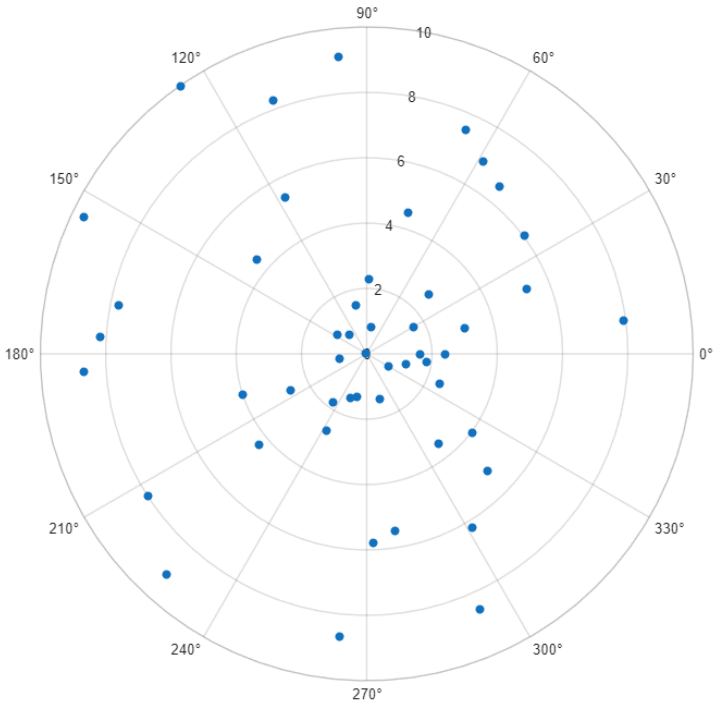
... (此部分与原版完全相同) ...

```
fprintf('\n================== 分析完□ ==================\n');
fprintf('在分析的 %d □数据中，共□□到 %d 个目□点。\n', length(frame_range),
length(detection_log));
log_output_file = fullfile(output_path, 'detection_log.mat');
if options.save_detection_log && ~isempty(detection_log)
    save(log_output_file, 'detection_log');
    fprintf('□□点□□信息已保存到: %s\n', log_output_file);
end

% --- 根据□□□用不同的□□函数 ---
if ~isempty(detection_log)
    switch plot_option
        case 'ppi'
            h_fig = fun_plot_cumulative_detections(detection_log, params,
NorthAngle, FixAngle);
            image_output_file = fullfile(output_path,
'cumulative_ppi_plot.png');
        case 'rdm'
            h_fig = fun_plot_cumulative_rdm(detection_log, params);
            image_output_file = fullfile(output_path,
'cumulative_rdm_plot.png');
        case 'az_range'
            h_fig = fun_plot_cumulative_az_range(detection_log, params,
NorthAngle, FixAngle);
            image_output_file = fullfile(output_path,
'cumulative_az_range_plot.png');
        case 'summary'
            h_fig = fun_plot_summary_dashboard(detection_log, params,
NorthAngle, FixAngle, velocity);
            image_output_file = fullfile(output_path,
'summary_dashboard.png');
        otherwise
            warning('未知的□□□□: %s。不□行□□。', plot_option);
            h_fig = [];
    end

    if options.save_cumulative_image && ishandle(h_fig)
        try
            saveas(h_fig, image_output_file);
            fprintf('累□□果□已保存到: %s\n', image_output_file);
        catch ME
            warning('无法保存□像。□□信息: %s', ME.message);
        end
    end
end
disp('所有流程□行完□。');
```

```
================== 分析完□ ==================
在分析的 152 □数据中，共□□到 0 个目□点。
所有流程□行完□。
```



--- 定□回□函数 ---

```
function updateWaitbar(h_waitbar, total_count)
    % 从□形句柄的 UserData 中□取并增加□数
    progress = h_waitbar.UserData + 1;
    h_waitbar.UserData = progress;

    % 更新□度条
    waitbar(progress / total_count, h_waitbar, sprintf('已□理 %d/%d □',
progress, total_count));
end
```

```
theta = linspace(0, 2*pi, 50);
rho = rand(1, 50) * 10;
polarscatter(theta, rho, 'filled'); % □制一个□□的极坐□散点□
```

Published with MATLAB® R2025a