Part1 describe the main workflow in next()

Calculate the relative pose T12 ->
if take key frame ->
Project landmarks to get the projected points and corresponding projected track ids ->
initializet the stereo match relationships data struct md_stereo ->
Detect keypoints and descriptors in both left and right frame ->
assigns T12 to the initial relative pose in md_stereo ->
Compute Essential matrix using T12 ->
Find matches between two frames using descriptors->
Find the match inliers in md_stereo adding epipolar constraint ->
Initial LandmarkMatchData md ->
Find matches of the landmarks in md->
Localize the camera to get the match inliers of the md->
Adding new land marks for the right camera frame ->
According to the max_num_kfs to remove old keyframes->
Optimize ->
Update image views ->
Process next frame.

Else ->
Project landmarks->
Detect keypoints and Descriptors between the two new frames ->
Find matches landmark ->
Localize camera ->
If md.inliners not sufficient, and opt is not running, and opt is not finished ->
set take keyframe true ->
after opt is not running and opt is finished ->
copy of cameras for opt in parallel thread->
opt.thread join ->
Process next frame

Part2 functionality of opt finished and opt running?

It will adjust the order of the optimization, if the optimization in take_key_frames is still running, then the optimization in else process will not be triggered. If removed, there might be issues, simultaneously optimizing causing issues.