

# Multimodal Large Model+robotic arm sorting (Voice Version)

Before running the function, you need to close the App and large programs. For the closing method, refer to [4. Preparation] - [1. Manage APP control services].

## 1. Function Description

After the program runs, wake up the voice module and input the color block sorting sequence by voice. The large model will plan the steps to complete the color block sorting task, then the program will control the robotic arm to sort the color blocks in sequence and place them at the set positions.

## 2. Startup

Users with Jetson-Nano mainboard version need to enter the docker container first and then input the following command. Users with Orin mainboard can directly open the terminal and input the following command:

```
ros2 launch largemodel largemodel_control.launch.py
```

After waking up the module, input by voice:

```
Sort the color blocks on the desktop in the order of yellow, red, blue, green
```

The program will sort according to the given order of yellow, red, blue, green. Before sorting the yellow color block, it must first detect whether it is in the removal list. If it is, it will directly grip from the removal list and place it at the yellow color block placement point; if not, it will determine whether there are other color blocks on top of the yellow color block. If there are, it will first remove the other color blocks on top of the yellow color block. If not, it will directly grip and place it at the yellow color block placement point. Other color blocks are also sorted according to this execution order until the task ends.

## 3. Task Planning

1. Call `check_remove(color)` function to check if the current target color block to be sorted exists in the removal list (where `color` is the color of the current target color block to be sorted, with values `red`, `blue`, `green` or `yellow`);
2. If the current target color block to be sorted exists in the removal list, perform the following operations:
  - (1) Call `grasp_from_rm_list(color)` function to grip the current target color block to be sorted from the removal list (meaning of `color` parameter is the same as above);
  - (2) Call `arm_stack(color)` to place the target color block at the specified position (meaning of `color` parameter is the same as above).
3. If the target color block does not exist in the removal list, the current color block to be sorted needs to be sorted according to the following steps (only output one action function at a time):
  - (1) Call `seewhat()` function to find the current target color block to be sorted;
  - (2) Call `seewhat()` function to determine if there are other objects on top of the current

target color block to be sorted;

(3) If there are no other objects on top of the current target color block to be sorted, perform the following steps (only output one action function at a time):

① Call `grasp_obj(x1, y1, x2, y2)` function to grip the current target color block to be sorted (where `(x1, y1, x2, y2)` are the top surface bounding box coordinates of the current target color block to be sorted);

② Call `arm_stack(color)` to place the current target color block to be sorted at the specified position (meaning of `color` parameter is the same as above).

(4) If there are other objects on top of the current target color block to be sorted, perform the following steps to first remove the object and then sort the current target color block, here only one action function can be output at a time:

① Call `remove_obj(x1, y1, x2, y2, color)` function to remove this object (where `(x1, y1, x2, y2)` are the top surface bounding box coordinates of this object, `color` is the color of this object, with values `red`, `blue`, `green` or `yellow`);

② Call `seewhat()` function to observe the environment;

③ If there are no other objects on top of the current target color block to be sorted, call `grasp_obj(x1, y1, x2, y2)` function to grip the current target color block to be sorted (where `(x1, y1, x2, y2)` are the top surface bounding box coordinates of the current target color block to be sorted);

④ Call `arm_stack(color)` to place the target color block at the specified position (meaning of `color` parameter is the same as above).

## 4. Core Code Analysis

You can refer to the content in **4. Core Code Analysis** from tutorial [17. AI Model - Text Version] - [Multimodal Large Model+robotic arm sorting]. The voice version and text version have the same action functions, only the task command input method is different.