iNexBot

Vision Process

Catalogue

Vision Process

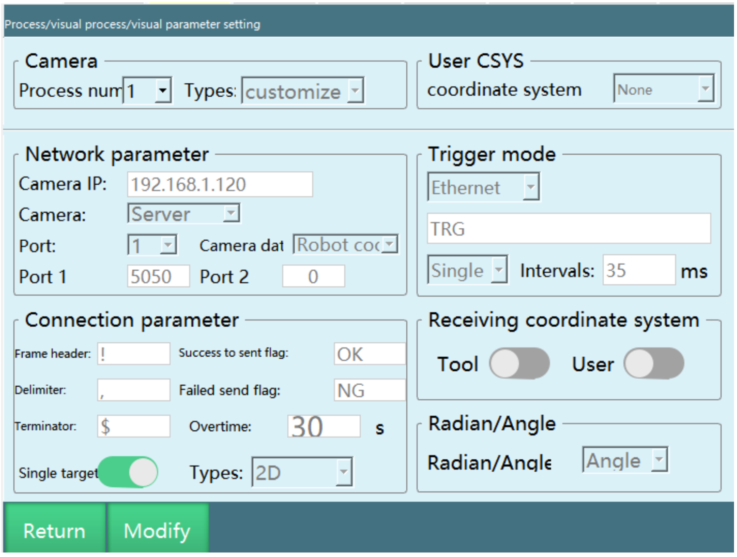
Vision process

It is not difficult for an industrial robot to perform repetitive motions, but if faced with a disordered environment, it can utilize the vision process to identify, analyze and judge the environment. The vision process has the advantages of precise positioning, convenient deployment, easy to use, flexible parameter configuration methods, and rich instructions. We often combine it with the conveyor belt process, which can effectively reduce the tracking error rate by using simple programming.

Vision parameter setting

Select "Vision process" and complete the settings of vision parameters/vision position/vision range/position debugging

The vision parameters are set as follows:



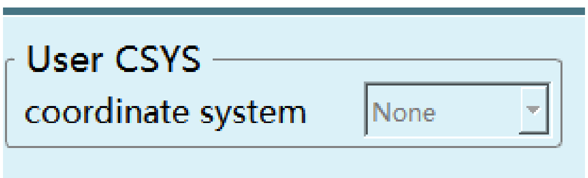
Camera selection

Process number: 1-99 process numbers are provided, each of which stores all the vision parameters and vision position parameters below that process number.

Type: Currently, only "Custom" type is supported, and users can set the parameters according to their needs.

User coordinate system

The vision system supports mapping the vision points to the user coordinate system, i.e. the points sent by the camera are the points in the vision coordinate system. Here you need to select a user coordinate system that has been matched with the camera.



If you select "None", the default is that the camera sends the points in a Cartesian coordinate system; the user can also select their own calibrated user coordinate system (the selected user coordinate system is calibrated in the "Settings - User coordinate calibration" page)

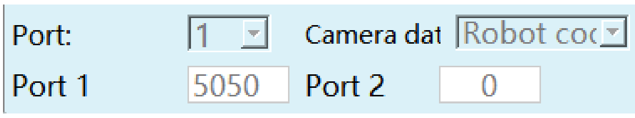
Network parameters

Camera IP: If the camera is used as the vision server, fill in the IP address of the camera here. The IP address of the camera and the IP address of the controller should be consistent with the first three bytes (counting from left to right), and the last one should be different, for example, both use the same network segment: 192.168.1.xxx.

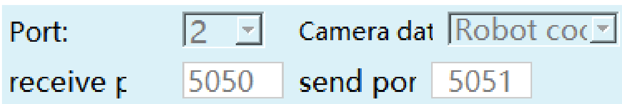
Camera: "Client" and "Server" can be selected here. If the camera is selected as the client, then the controller is the server, and the camera needs to be actively connected.

Camera data: Here you can choose "Robot coordinates" and "Pixel coordinates". If you choose "Robot coordinates", the data sent by the camera is the coordinates of the robot; if you choose "Pixel coordinates", the data sent by the camera is the pixel coordinates in the camera coordinate system

Number of ports: If the vision server uses the same port for data sending and receiving, the number of ports is 1



If different ports are used for data sending and receiving, then the number of ports is 2



Port 1 is to receive data; port 2 is to send data (port number cannot be set to the same value)

Port 1: Generally use 5050

Connection parameters

The frame header, separator and terminator cannot be set to the same character at the same time

The frame header and terminator can be set to blank except for the separator

Frame header: The beginning of the signal transmission. This must be the same as the parameter of the camera configuration.

Separator: Used to separate multiple signals when transferring them. This must be the same as the parameter of the camera configuration. **(This cannot be set to null)**

Terminator: The symbol that determines the end of signal transmission. This must be the same as the parameter of the camera configuration.

Successfully sent identifier: After the camera has taken the photo and successfully identified it, a success identifier will be sent after sending.

Failed to send identifier: If the camera has taken the photo but failed to identify it, a failure identifier will be sent.

Note: The above parameters are user-defined

For example, set the frame header to blank, set the separator to ",", set the terminator to "$", and turn on the "Single target" enable switch

The data format is: ,X,Y,Rz,$

Timeout: When this time is exceeded, it is determined that the connection has timed out and the connection is stopped. When it is filled as 0, there is no limit.

Single target : Turn on the enable switch, the camera will identify only one target point at a time.

Type: 2D, 2D+Height, 3D; e.g. the camera sends a string (frame header "**!**", separator "**,**", terminator "**$**"):

2D: The data format is: **!,X,Y,Rz,$**

2D+Height: The data format is: **!,X,Y,Rz,h,$**

3D: The data format is: **!,X,Y,Z,A,B,C,$**

Turn off the"Single target" enable switch: More than one target point can be identified, and the example N represents the number of identified target positions.

Type: 2D, 2D+Height, 3D; e.g. the camera sends a string (frame header "**!**", separator "**,**", terminator "**$**"): the N represents the number of identified target positions.

2D: The data format is: **!,N,X,Y,Rz,X,Y,Rz,$**

2D+Height: The data format is: **!,N,X,Y,Rz,h,X,Y,Rz,h,$**

3D: The data format is: **!,N,X,Y,Z,A,B,C,X,Y,Z,A,B,C,$**

Trigger method

I/O: Give the camera a trigger signal through the I/O board, here you need to set the DIN (IO input) signal port of the I/O.

Ethernet: Generally, the default trigger method is Ethernet. When the camera receives the "TRG" (or user-defined string) here, it should reply the coordinate value to the controller.

Trigger condition

Single trigger: When the condition is single trigger, the camera will be triggered once each time you run the VISION_TRG instruction in the program.

Continuous trigger: When the condition is continuous trigger, the camera will be triggered continuously each time you run the VISION_TRG instruction in the program.

Interval time: The time interval during continuous triggering (triggering cycle);

Receiving coordinate system

The received point information is the point information sent by the camera with a specific tool hand under a specific user coordinate system

Tool: When this enable switch is turned on, the point sent by the camera contains the tool hand used (used when working with multiple tool hands)

User: When this enable switch is turned on, the point sent by the camera contains the user coordinate system used (used when there are multiple workbenches)

Note: Before turning on the "Tool" and "User" enable switches, the "Hand-eye calibration user coordinate system" cannot be "None" (If both are off, it needs to be set to "None"), and the "User" and "Tool" enable switches can be turned on/off at the same time.

Angle/radian setting

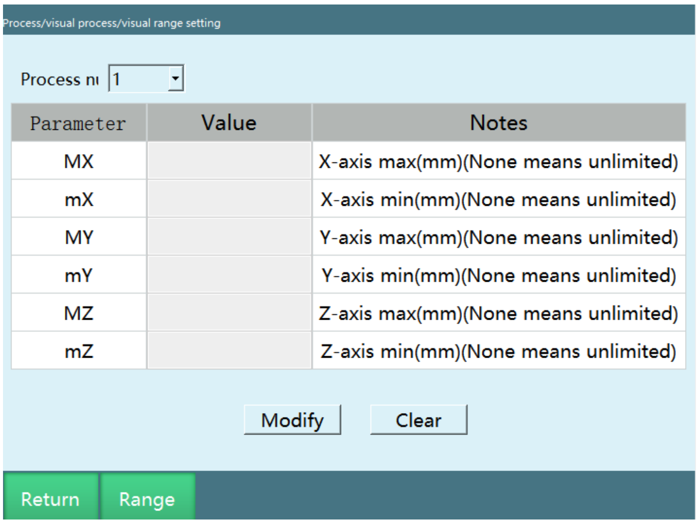
Select the unit type for the A/B/C axis in the "Vision position parameters", the unit of radian is rad, and the unit of angle is º (degree).

Note: The setting of the angle/radian affects the content of the analytical data and has nothing to do with the angle/radian switch in the operation

parameter. The angle/radian setting in the operation parameter only affects the display of the content about the angle/radian on the teach pendant

Vision range setting

Enter the "Vision range setting" interface from "Process" - "Vision process" - "Vision range setting".

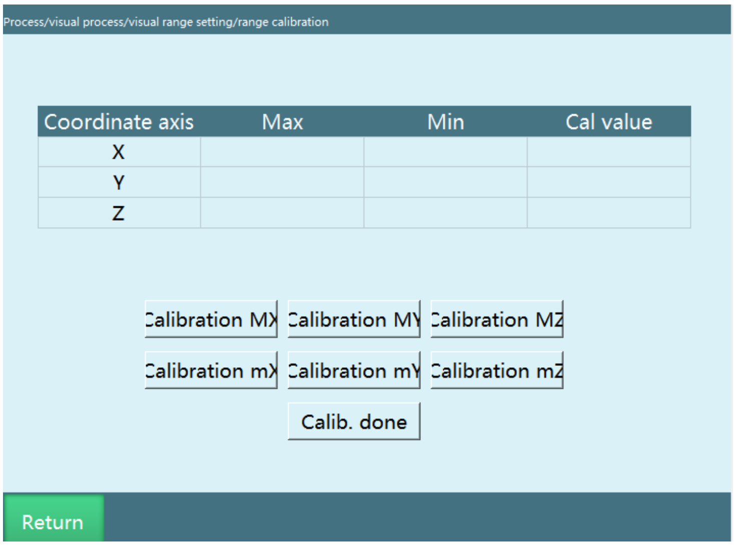


In order to avoid that the address parameters returned by the camera exceed the range that the robot can reach, the maximum range that the robot can reach is specified. If the parameters returned by the camera are out of range, the data will be automatically filtered, and the data will not take effect.

You can complete the calibration by manual teach or fill in the corresponding values directly.

Process number: 1-99 process numbers are provided, each of which stores all the vision range parameters below that process number.

Range calibration: Calibrate the maximum and minimum values of the XYZ axes in the Cartesian coordinate system



Calibrate MX: Calibrate the X-axis maximum value

Calibrate mX: Calibrate the X-axis minimum value

Calibrate MY: Calibrate the Y-axis maximum value

Calibrate mY: Calibrate the Y-axis minimum value

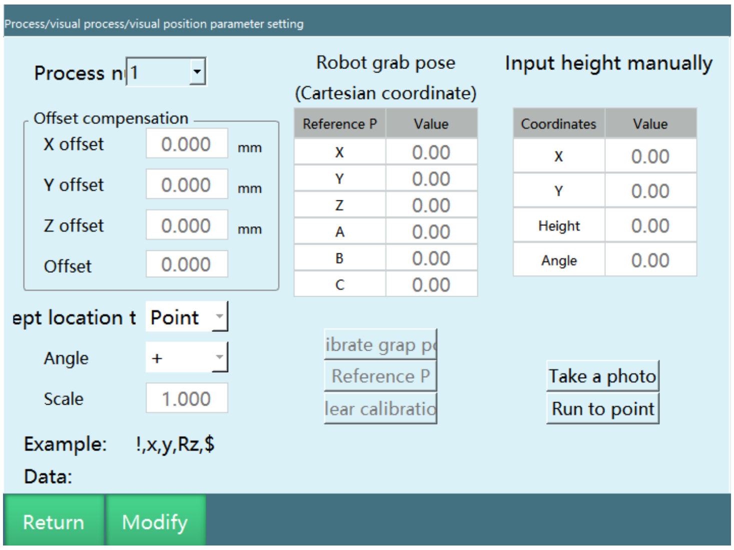
Calibrate MZ: Calibrate the Z-axis maximum value

Calibrate mZ: Calibrate the Z-axis minimum value

Calibration completed: Click to record all calibrated values in the maximum and minimum values.

Vision position parameters

Enter the "Vision position parameters" interface from "Process" - "Vision process" - "Vision position parameters".



Offset compensation: If the robot's gripping position is offset from its actual position in a fixed direction every time, please fill in the compensation here and it will be automatically compensated to the correct position.

Scale factor: If the position value sent by the camera is reduced by a specific ratio, you need to fill in the scale factor here. For example, if the camera sends a value of (300,200,100) and the actual position is (3,2,1), then 0.01 should be entered here.

Calculation formula: scale factor = actual position value/position value sent by camera

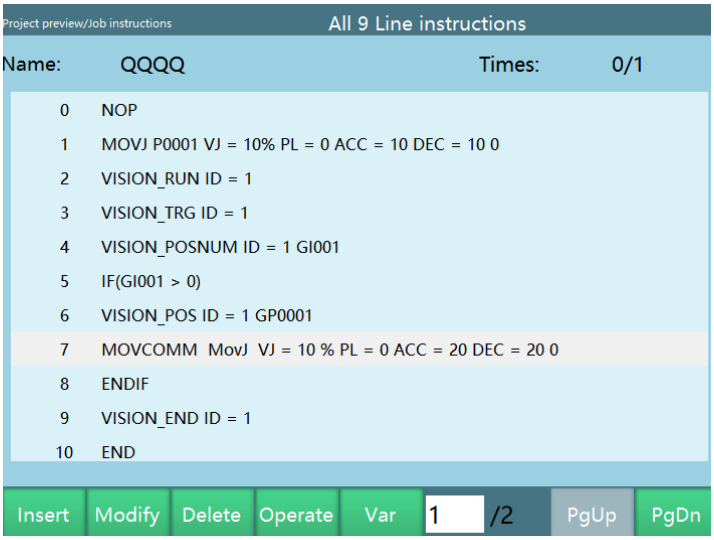
Angle direction: The direction of the points sent by the camera can be the same as or opposite to the direction of the rotation angle of the robot

Receiving position type: Point/trajectory

When selecting "Point", the camera takes photos and sends the points to the controller

When selecting "Trajectory", the camera will identify the trajectory and send a series of points during dispensing or spraying, **and the trajectory is run through an external point instruction**. If "Trajectory" is selected, **"Single target"** in the "Vision parameter setting" will only identify one segment of trajectory, if "Single target" is turned off, then the camera will identify multiple segments of trajectory

When "Trajectory" is selected for the "Receiving position type", the program job file is as follows:



Reference point and height

Calibrate gripping attitude: Here you need to mark the end attitude of the robot when gripping the object. After the calibration, every grip will be done with this attitude. (**The XYZ values here do not affect the position at the time of gripping**)

Run to reference point: Run to the point that was calibrated when the gripping attitude was calibrated.

Clear calibration: Clear the point data used to calibrate the gripping attitude.

Camera coordinates: If the camera cannot send the gripping height, you need to fill in the gripping height Z in the table on the right. If the camera can send the gripping height, the setting here has no effect. After setting, press and hold the DEADMAN button to power on, click the [Try photo] button to take a photo for test, the data sent by the camera will be displayed at the "Camera coordinates" and "Receiving data" parts. Press and hold the DEADMAN button to power on after taking a photo, click the [Run to this point] button, move the robot to the photo-taking position to verify whether it is accurate.

Try photo: Power on the servo, click "Try photo", open the network connection, and send the data according to the sample format

Sample format: Verify the arrangement according to the connection parameters already set in the "Vision parameter setting". For example, if the frame header is W, the separator is # and the terminator is $ in the connection parameters, and the height information is sent, then the format is W#x#y#angle#h#$

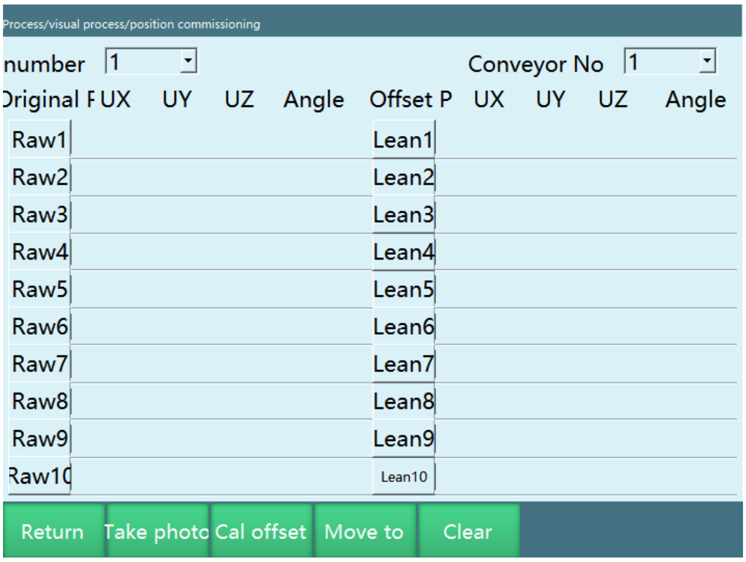
Receiving data: W#x#y#angle#h#$

Run to this point: The robot moves to the position sent by the camera

Position debugging

It is used in combination with the conveyor belt process for the debugging of the conveyor belt. After the camera takes a photo, it will send a point data to be stored in **[Original point]**, but the workpiece will be taken out by the conveyor for a distance, click "Calculate offset", the calculated position after offset will be stored in [**Point after offset**], click [Move here], and the robot will go directly to the calculated position after offset.

Enter "Process" - "Vision process" - "Position debugging", use vision process plus conveyor tracking process to debug the conveyor belt



Process number: The process number of the vision process.

Conveyor process number: The process number of the conveyor belt to be debugged.

Take photo: Press and hold the DEADMAN button to power on and click the [Take photo] button to take a photo for test, the position data sent by the camera will be displayed at [**Original point**] on the left side.

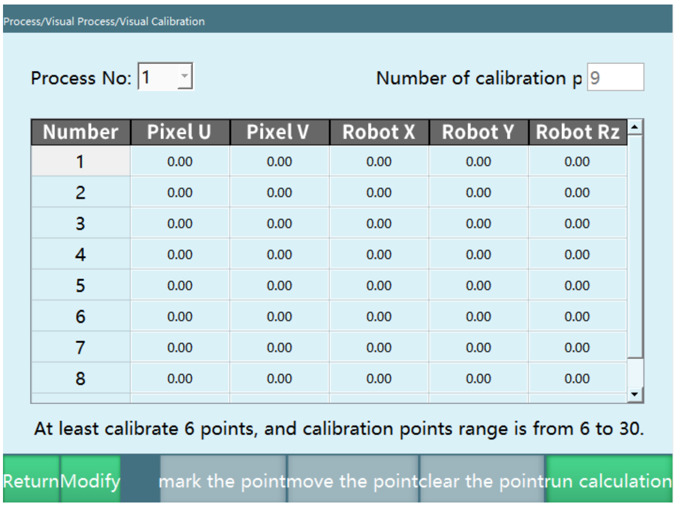
Move here: After taking a photo, press and hold the DEADMAN button to power on, select the corresponding point and click the [Move here] button, the robot will move to the position sent by the camera.

Calculate offset: After taking a photo, start the conveyor belt so that the workpiece is transported for a certain distance, and click "Calculate offset", the [**Point after offset**] on the right side will display the workpiece point after offset.

Clear: Clear all points.

Vision calibration

Vision calibration refers to installing the camera on the tool hand, marking the current point information of the robot, running the calculation, triggering the camera to go to the current point of the robot, and obtaining the corresponding pixel data. When all pixel data are obtained, the conversion relationship between camera data and robot points is calculated. The camera points sent subsequently can be converted into the actual motion points of the robot through the conversion relationship.



Process number: Process number of the vision process

Number of calibration points: The number of points to be taught; the range of the number of calibration points is 6-30

Mark this point: Record the current robot point data

Move here: Press and hold the DEADMAN button to power on the robot, select the serial number and click the [Move here] button, the robot will move to the point marked by the serial number

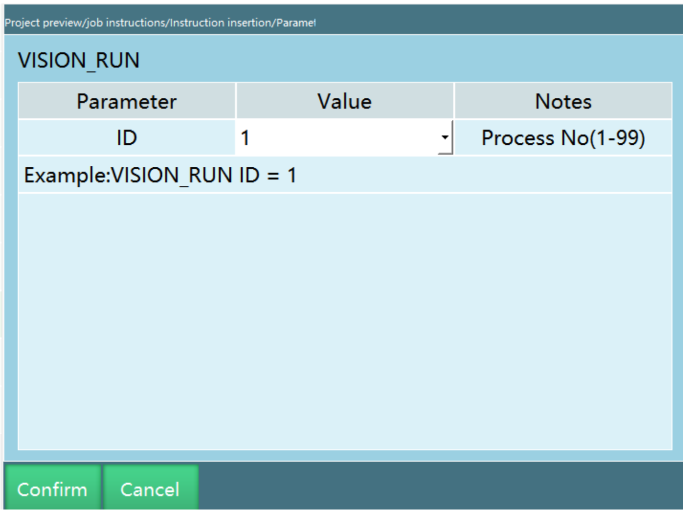
Clear this point: Clear the point data of the selected serial number, but do not clear the pixel data, the pixel data will be overwritten by the newly captured data after running the calculation

Run calculation: Click "Run calculation", the robot will move according to the point data taught before, and each time it moves to a point, it will trigger a photo taking and record the current pixel data

Vision instructions

VISION_RUN

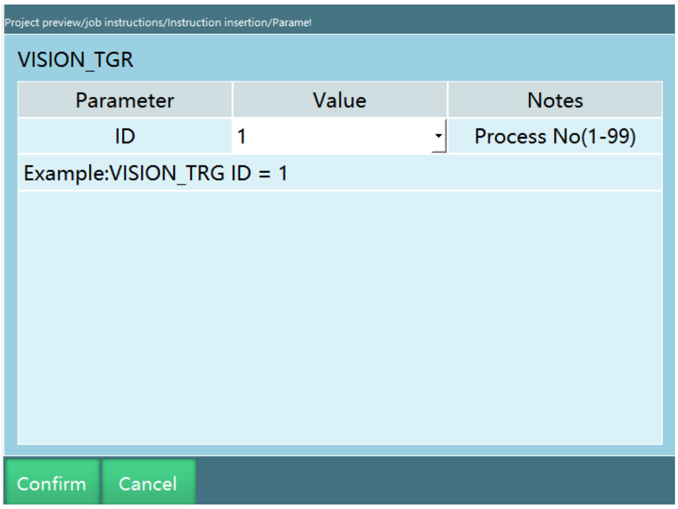
Vision start instruction



Run this instruction to connect the controller and the camera

VISION_TRG

Vision trigger instruction



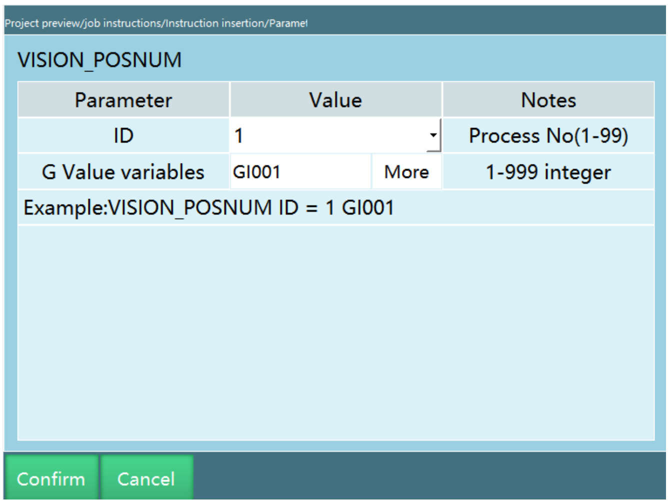
The controller issues a trigger signal after running this instruction

The specific trigger method is set in the "**Vision process-Vision parameter setting"** interface: 1. Select "IO": run this instruction to send out the corresponding IO signal; 2. Select "Ethernet": run this instruction to send a custom string to the camera.

After running this instruction, it will wait for the return value (position data sent) from the vision server, and continue to run the next instruction after obtaining the position.

VISION_POSNUM

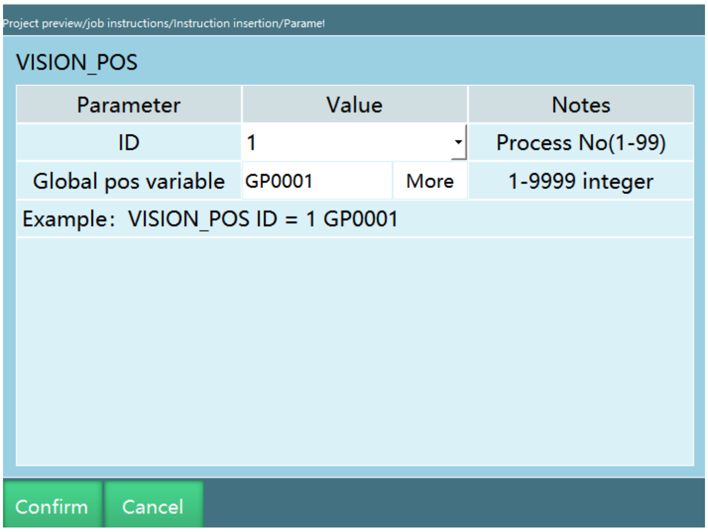
The instruction to get the number of vision positions



When the "Single target" enable switch is turned off in [Process-Vision process-Vision parameter setting], record the number of points sent by the camera. Every time this instruction is run, the number of points will be reduced by one

VISION_POS

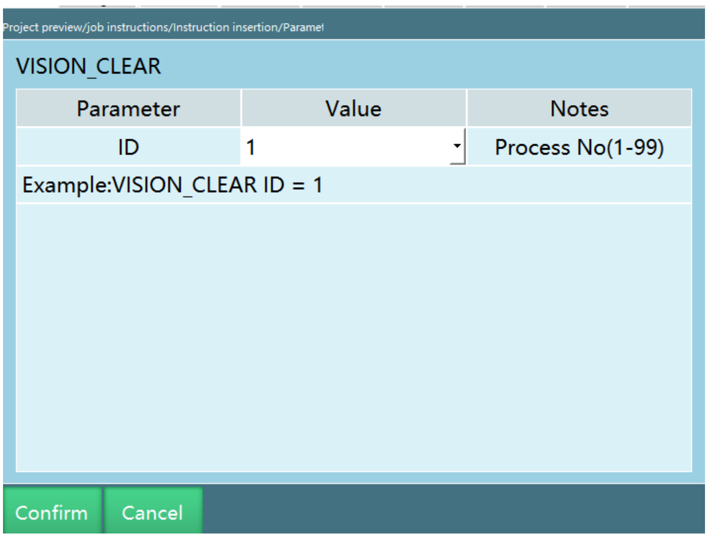
The instruction to get vision position



The point information sent by the camera is successively cached in the position of GP0001, for example, when two points are sent, the first time the instruction is run, the GP0001 stores the point 1, and the second time the instruction is run, the GP0001 stores the point 2

VISION_CLEAR

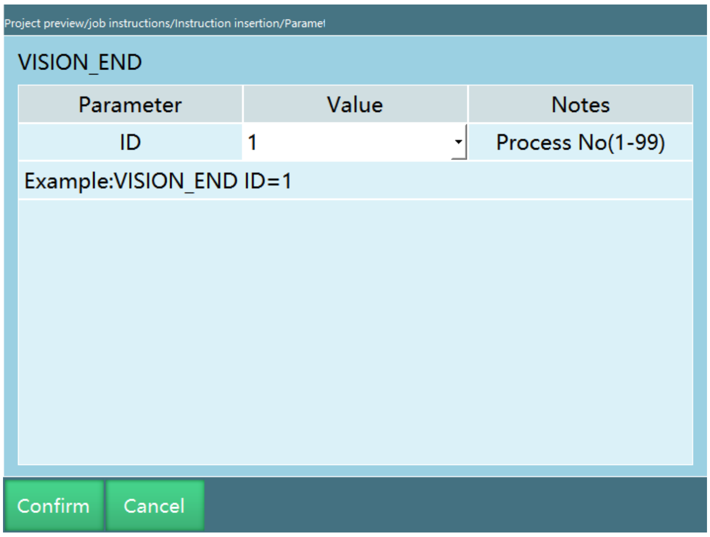
The instruction to clear vision position information



The cleared point information is the point sent by the camera in the **Vision position parameter**

VISION_END

Vision end instruction



End the vision process and the controller is disconnected from the camera.

Use cases

Gripping application

After the camera takes a photo of the material, it sends the data to the robot, then the robot goes to grip the material

Programming: