



FINGERPRINTS

NAVIGATION TEST APP USER GUIDE

WWW.FINGERPRINTS.COM

Fingerprint Cards AB • Visiting address: Kungsporsplatsen 2 • P.O. Box 2412 • SE 403 16 Gothenburg • Sweden • +46 31 60 78 20 • VAT SE556154238101





TABLE OF CONTENTS

NAVIGATION TEST APP USER GUIDE	1
OVERVIEW	3
INSTALLATION	3
DEMO MODE	3
TEST MODE	4
LOGS	6
SETTINGS	7
TROUBLESHOOTING && TWEAKING	9
NAVIGATION EVENT	9
SENSE TOUCH	9
APPENDIX A	11
APPENDIX B	12
Revision history	12





OVERVIEW

The navigation test application allows you to demo, test, and tweak performance of the Fingerprints navigation feature. The application is divided into 4 logical sections where the various functionality can be accessed:

- Demo Mode
- Test Mode
- Logs
- Settings

INSTALLATION

To install the navigation test application, you first need Android Debug Bridge (ADB) installed on your local machine. Once installed, simply run the 'adb install' command and pass the navigation test Android Application Package (APK) as argument:

```
$ adb install -r NavigationTest.apk
```

For the application to work properly Fingerprints software (SW) must also be built with the correct flags

- FPC_CONFIG_NAVIGATION=1 to enable the navigation feature.
- FPC_CONFIG_ENGINEERING=1 to enable the extension interface that the application uses to communicate with the Fingerprints SW.
- FPC_CONFIG_FORCE_SENSOR=1 to enable HW Sense Touch support. (optional)
- FPC_CONFIG_NAVIGATION_FORCE_SW=1 to enable SW Sense Touch support. (optional)

DEMO MODE

Demo mode works like a graphical interface for navigation input events and displays the active event on screen.

There is an option in Demo that allows you to turn on/off demoing of Sense Touch, that is, how much pressure that is applied to the sensor.

- Sense Touch

Will enable display of active force reading in form of a pressure bar. In case of HW Sense Touch being enabled, a Force Press icon will also be displayed when the configured threshold has been met. But this function is not supported by most of FPC sensors.



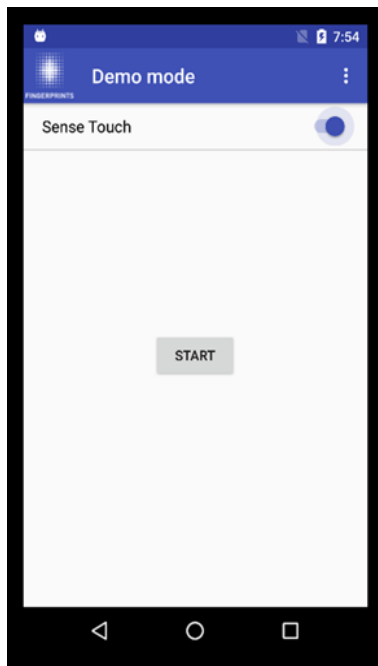


Figure 1 Enable force reading

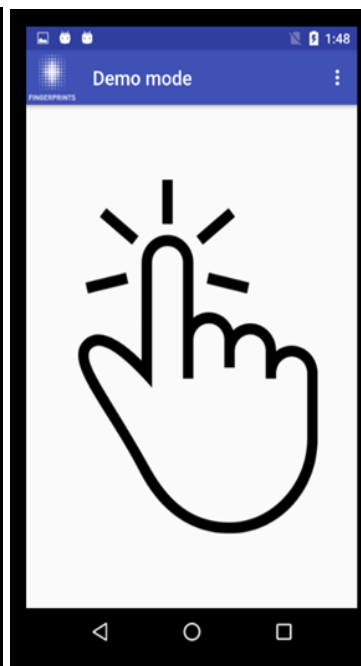


Figure 2 Registered single click

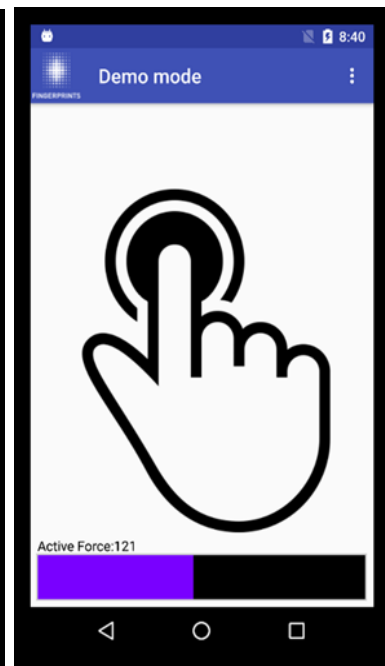


Figure 3 Registered force press and showing the active force

Once suitable options have been selected simply press the start button and you will be taken to an interface that displays navigation input on the screen. See Appendix A for icon description.

TEST MODE

Test mode allows the user to test various input events in fixed or random order. This can be used to test and verify the accuracy of a subset of inputs.

In the test setup interface, user can input what type of events that should be tested and how many occurrences. For ease of use there is also a few quick buttons:

- **Add 1 TO All**

Adds one of each navigation event to the test scope.

- **CLEAR**

Removes all navigation events from test scope.

- **Randomize test sequence**

Randomizes the order of the navigation events during the test.



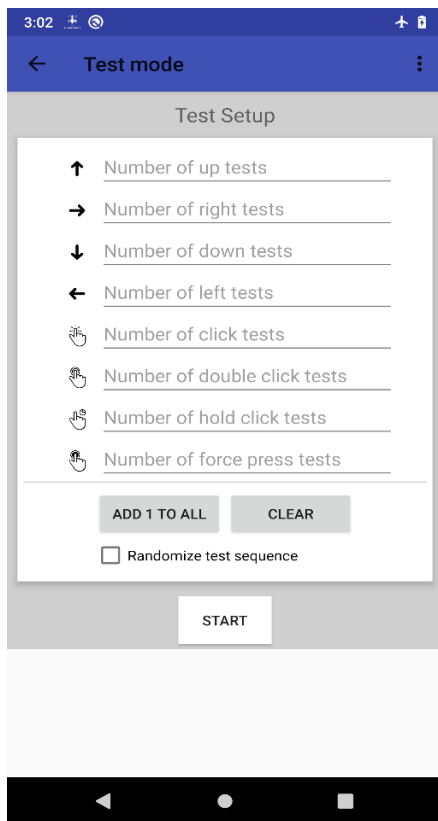


Figure 4 Empty test setup

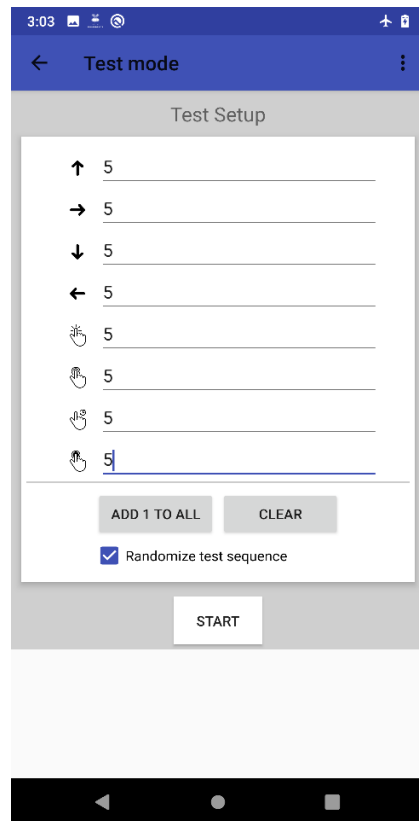


Figure 5 Setup to test 5 occurrences of each navigation event

Once you are satisfied with your test setup simply press start button at the bottom of the screen to begin the test run.

Once test mode has been started you will see an icon on screen, this is the expected input. Try inputting the navigation event that is displayed and the screen should flash either green (input successful) or red (input failed). After the input has been registered, the next event in the test sequence will be displayed.

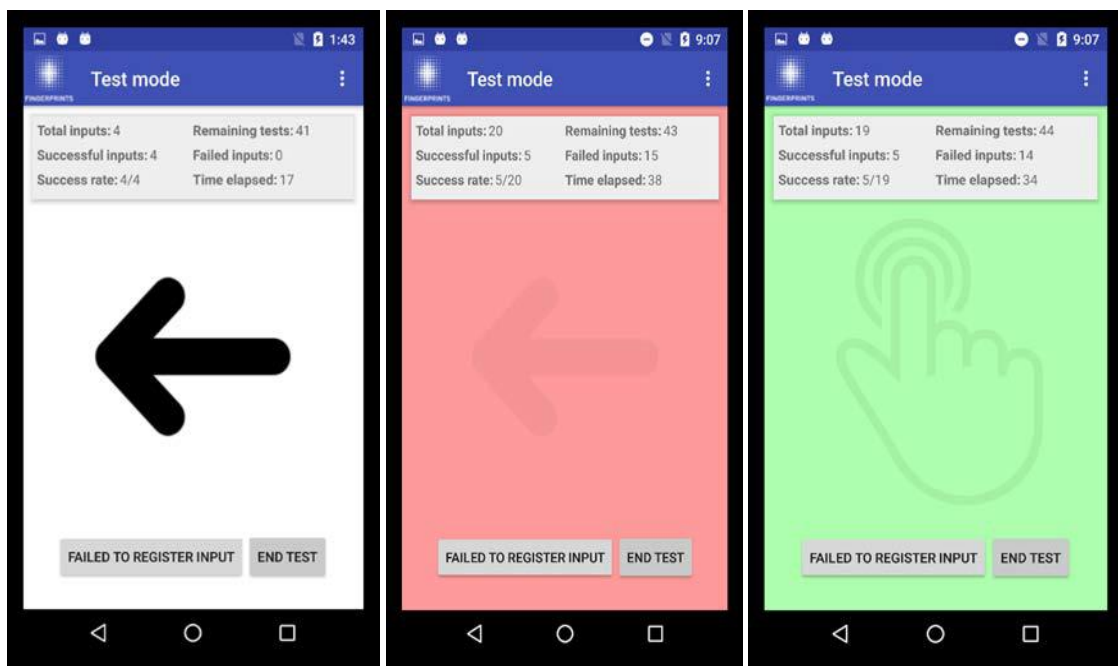




Figure 6 Waiting for left swipe

Figure 7 Failed to swipe left

Figure 8 Successful double click

As you continue with the test run you will see live results displayed on the top parts of the screen, if you wish to end the test run you can do so at any time by pressing the end test button. This will summarize all the input up until this point into a test report and display it on screen. If you wish to restart a test run you can simply do so by pressing the standard android back button which will take you back to the setup screen again with the same test setup you had previously given.

Once a test run has been finished by either finishing all test inputs or ending the test early with the end test button you will be taken to the test summary screen. Here you can get a quick overview of the results and if you wish, also see further details by clicking the Show log button. The log displays each test and the corresponding test input that was registered for that test. This combined with system log can be used to evaluate cause of successful or wrong input.

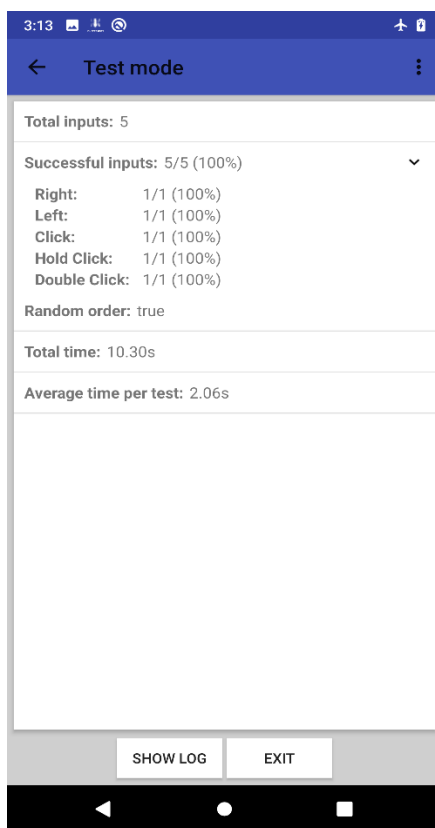


Figure 9 Test results screen

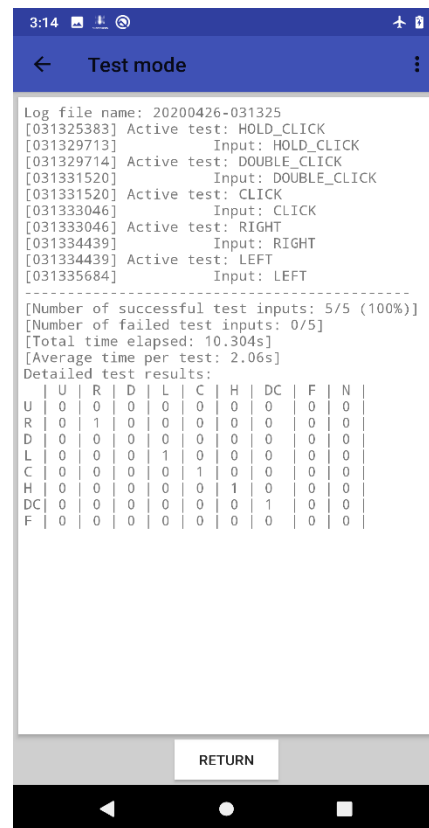


Figure 10 Show log screen

LOGS

In the logs section, you will find test result logs for all tests runs that is been completed in test mode. Each log will be timestamped and can easily be viewed by clicking on it in the interface. Once a log has been opened you can choose to delete it if no longer required.

If you wish to clear all logs you can simply do so by pressing the delete all logs button and confirm the action on the main log screen. All logs are stored locally on the device and can by default be found under:

/data/data/com.fingerprints.navigationtest/files/



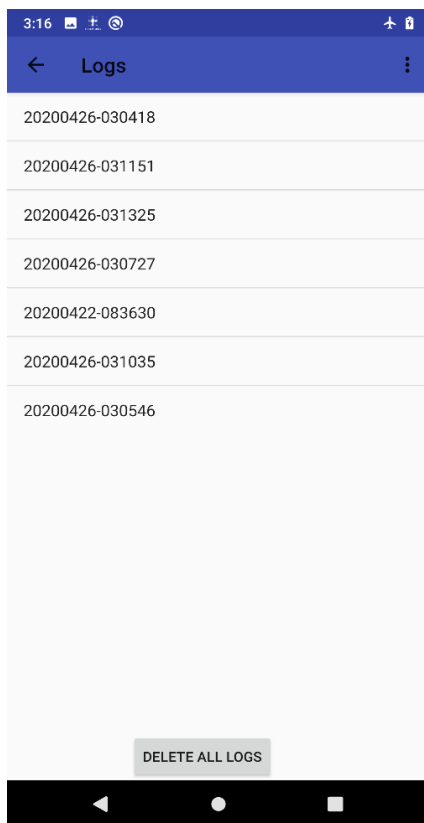


Figure 11 Logs section

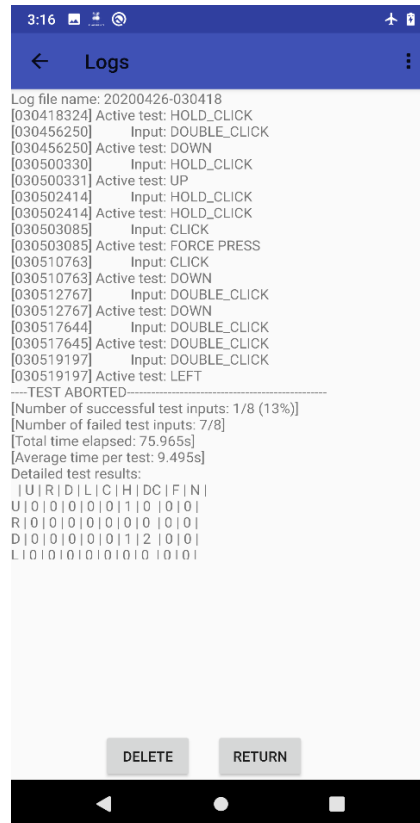


Figure 12 Opened log

SETTINGS

The settings section allows you to tweak and configure navigation parameters to achieve better performance, it contains several key features:

- Enabled/disable the navigation feature.
- Get currently active navigation configuration.
- Edit the active navigation configuration.
- Reset to using the system default configuration.

When opening the setting UI of navigation app, the default configuration would be loaded in the UI (see Figure 13). To alter the navigation configuration, first disable navigation. To do this simply flick the enabled navigation switch to off. Once this has been done you will be able to select and modify the individual configuration parameter fields displayed in the interface. The input for all configuration parameters can only be up to 3 digits long (0-255), if it is out of data range, the navigation would not work. Once you have modified the settings just enable navigation again to apply the new configuration.



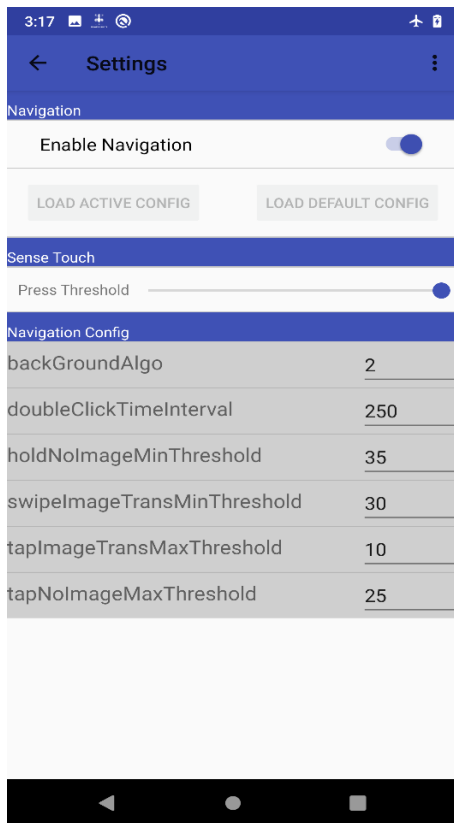


Figure 13 Settings navigation enabled

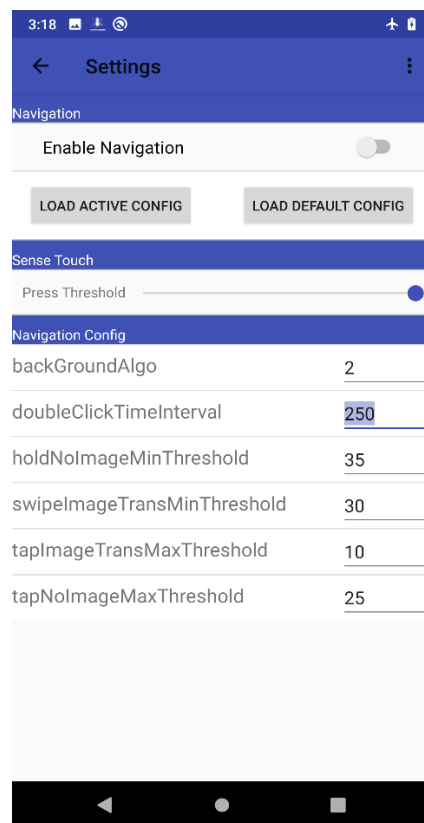


Figure 14 Editing active configuration

The get active configuration button can be used to fetch the currently used configuration. The active configuration is always loaded by default each time you enter the settings menu. The intended use case of the button is if you have disabled navigation and changed the setting but want to revert to the original settings.

The load defaults button can be used to revert to the system default configuration, this can for example be used if you have changed and applied a new configuration but later which to revert to the original system default configuration.

Name	Unit	Range	Description
backGroundAlgo	N/A	0-255	The enumerated value which indicating the mode of navigation ALGO, the value could be following: 1: AFD based navigation mode 2: DFD + image-based navigation mode 3: DFD + image-based navigation to give raw dx, dy 4: HW based navigation mode(most sensors unsupported).
doubleClickTimeInterval	ms (milliseconds)	0-255	Determines the time interval to wait for a second click after a single click has been detected. If a second click is detected within this time the motion will be considered a double click. If time exceeds this value a single click will be generated.



FINGERPRINTS

holdNoImageMinThreshold	Number of pictures	0-255	Minimum image frames a hold click should have.
swipeImageTransMinThreshold	px (pixels)	0-255	Minimum amount of movement needed to consider a swipe
tapImageTransMaxThreshold	px (pixels)	0-255	Maximum amount of movement in pixels a tap could move .
tapNoImageMaxThreshold	Number of images	0-255	Maximum image frames a tap could have .

Table 1 Navigation configuration overview

TROUBLESHOOTING & TWEAKING

If navigation is found to work less than adequate in a system, it is possible to manually tweak the configuration values. There are several factors to consider when tweaking a sensor for navigation in order to get optimal results, such as:

- Where the sensor is mounted?
- What orientation does the sensor have (mounted direction)?
- How large is the sensor?
- Variation in hardware and TA performance
- How does the user interact with the sensor?

These factors may all play a role in how you configure the sensor for optimal results. An example of this could be if the sensor is mounted on the top side or back it usually becomes harder to swipe up than any other direction, in that case you might therefore consider to have the up threshold set slightly lower comparison to other directions. The default configuration supplied from Fingerprints is tested on reference Nexus device that has back mounted sensors and uses the Qualcomm QSEE platform.

In the Settings section of the app it is possible to enable/disable navigation and change the configuration parameters for navigation via the Fingerprint Extension API. Note that any changes done via the app is only for test purposes and will not be permanent as values will automatically be reset to default during a device reboot.

To permanently change the settings, the default configuration in fingerprints software must be modified with the new values and rebuilt into a new binary.

Please consult your local fingerprints engineer to help with modifying the default configuration or if you are unable to find a configuration with satisfactory navigation performance.

NAVIGATION EVENT

The Fingerprints SW will report navigation events using the Linux input (via uinput driver). The android input system will in turn translate and deliver these events to the applications.

In Appendix B you can find the content of a reference `uinput-fpc.kl` file where each swipe direction is mapped to directional DPAD input and click, hold click, double click is mapped to BUTTON_A, BUTTON_B, BUTTON_C respectively.

You can easily remap which android event should be generated by each navigation event by updating the kl file, please see android key layout and key events for reference.

- **Note:** The navigation test application is implemented to listen to the specified android events listed in the kl file in Appendix B, so if these are changed then the app needs to be updated as well.

SENSE TOUCH

Sense Touch events are reported in similar manner as navigation events described in the section above. The raw force value is by default mapped to LTRIGGER in the reference `uinput-fpc.kl` file by using axis which simulates an analog trigger input.

- SW sense touch

WWW.FINGERPRINTS.COM





FINGERPRINTS

In SW Sense Touch, Fingerprints supplies the raw force value and it is up to the original equipment manufacturer (OEM) to implement soft/hard press according to their needs. Sense Touch events can be setup to be compatible with all other existing click events: single click, double click and hold click. The final customer solution for sense touch events can be done and integrated in any way see fit by using the raw data and adapting it to specific customer needs.

- **HW sense touch**

Most of our sensors do not support this function.

WWW.FINGERPRINTS.COM



APPENDIX A









Swipe Events	Click Events	Force Events
 <p>Swipe Up</p>	 <p>Single Click</p>	 <p>Force Press</p>
 <p>Swipe Right</p>	 <p>Double Click</p>	
 <p>Swipe Down</p>	 <p>Hold Click</p>	
 <p>Swipe Left</p>		

Table 3 Navigation event icon description





FINGERPRINTS

APPENDIX B

Fingerprints Key Layout File

Android Q:

/vendor/usr/keylayout/uinput-fpc.kl

Other:

/system/usr/keylayout/uinput-fpc.kl

File Content

key 108 DPAD_RIGHT

key 103 DPAD_LEFT

key 106 DPAD_UP

key 105 DPAD_DOWN

key 304 BUTTON_A

key 305 BUTTON_B

key 306 BUTTON_C

key 307 BUTTON_X

axis 2 LTRIGGER

Revision history

Rev	Date	Changes	Author
dev-2.30	2020-11-27	Adapt to the newest app	Dave Xu

WWW.FINGERPRINTS.COM

