

[API reference](#)[Gesture Handlers](#)[Force touch](#)

Version: 2.6.0 – 2.12.0

# ForceTouchGestureHandler (iOS only)

## DANGER

Consider using the new [gestures API](#) instead. The old API is not actively supported and is not receiving the new features. Check out [RNGH 2.0 section in Introduction](#) for more information.

A continuous gesture handler that recognizes force of a touch. It allows for tracking pressure of touch on some iOS devices. The handler [activates](#) when pressure of touch is greater or equal than `minForce`. It fails if pressure is greater than `maxForce`. Gesture callback can be used for continuous tracking of the touch pressure. It provides information for one finger (the first one).

At the beginning of the gesture, the pressure factor is 0.0. As the pressure increases, the pressure factor increases proportionally. The maximum pressure is 1.0.

The handler is implemented using custom [UIGestureRecognizer](#) on iOS. There's no implementation provided on Android and it simply renders children without any wrappers. Since this behaviour is only provided on some iOS devices, this handler should not be used for defining any crucial behaviors. Use it only as an additional improvement and make all features to be accessed without this handler as well.

## Properties

See [set of properties inherited from base handler class](#). Below is a list of properties specific to `ForceTouchGestureHandler` component:

### `minForce`

A minimal pressure that is required before handler can [activate](#). Should be a value from range `[0.0, 1.0]`. Default is `0.2`.

### `maxForce`

A maximal pressure that could be applied for handler. If the pressure is greater, handler fails. Should be a value from range `[0.0, 1.0]`.

### **feedbackOnActivation**

Boolean value defining if haptic feedback has to be performed on activation.

## Event data

See [set of event attributes from base handler class](#). Below is a list of gesture event attributes specific to `ForceTouchGestureHandler`:

### **force**

The pressure of a touch.

## Static method

### **forceTouchAvailable**

You may check if it's possible to use `ForceTouchGestureHandler` with `ForceTouchGestureHandler.forceTouchAvailable`

## Example

See the [force touch handler example](#) from Gesture Handler Example App.

```
<ForceTouchGestureHandler
  minForce={0}
  onGestureEvent={this._onGestureEvent}
  onHandlerStateChange={this._onHandlerStateChange}>
  <Animated.View
    style={[
      styles.box,
      { transform: [{ scale: Animated.add(1, this.force) }] },
    ]}
  />
</ForceTouchGestureHandler>
```

