

Image

基本用法:

```
<Image source={require('./my-icon.png')} />
```

為了提供高解析度，需要作一個目錄結構出來，高解析度的手機會自動使用:

```
.
├── button.js
└── img
    ├── check@2x.png
    └── check@3x.png
```

```
<Image source={require('./img/check.png')} />
```

注意事項: 保持圖片的名稱，不要用組合名稱的方式

```
// prettier-ignore
// 好的語法
<Image source={require('./my-icon.png')} />

// 不好的(錯誤的)作法
var icon = this.props.active ? 'my-icon-active' : 'my-icon-inactive'
<Image source={require('./' + icon + '.png')} />

// GOOD
var icon = this.props.active
  ? require('./my-icon-active.png')
  : require('./my-icon-inactive.png')
<Image source={icon} />
```

注意事項，如果要使用寬高動態的情況下，要手動設定 { width: undefined, height: undefined } 在 style 屬性上。其他時候應該是自動寬高。

圖片格式支援

目前預設支援的圖片格式如下:

png, jpg, jpeg, bmp, gif, webp (Android only), psd (iOS only)

註: GIF 與 WebP 預設在 Android 是不支援的，需要額外再設定

Images 來自 Hybrid App 的資源中

圖如果是放在 Xcode 裡或是 Android 裡，要用 Hybrid 的方式。但這方式不會作安全檢查，需要由開發者自行檢查與手動加寬高。(圖片會存在 Xcode asset catalogs 或 Android drawable 資料夾中):

```
<Image source={{ uri: 'app_icon' }} style={{ width: 40, height: 40 }} />
```

網路圖片

注意: 需要自行加上寬高。 注意: 要用 https，不然有時候會失連或失效。

```
// 好的作法
<Image
  source={{uri: 'https://facebook.github.io/react/img/logo_og.png'}}
  style={{width: 40, height: 40}}
  alt="Facebook logo"
  data-bbox="58 925 534 964"/>
```

```
    style={{width: 400, height: 400}}
  />

// 不好的(錯誤的)作法
<Image source={{uri: 'https://facebook.github.io/react/img/logo_og.png'}} />
```

其它小抄

SVG 支援

- Expo 已內建 react-native-svg

有很多套件可以用於 svg 的圖檔，例如：

- react-native-svg-transformer
- react-native-remote-svg
- react-native-svg-uri

遠端圖片無法顯示的解決方式

1. 使用 ios 模擬器選單來重新整理: Simulator -> Reset content and settings
2. 網址記得使用 HTTPS，例如: <https://i.imgur.com/UePbdph.jpg>

參考: <https://github.com/facebook/react-native/issues/289>

手機上的照片存取

React Native 詳見 CameraRoll 的說明，主要針對 iOS。

<https://facebook.github.io/react-native/docs/cameraroll>

Expo 另外提供了 ImagePicker 方法，可以以異步的方式來存取手機上的照片庫。

<https://docs.expo.io/versions/latest/sdk/imagepicker/>

Expo 另外提供了 Camera 方法，可以以異步的方式來存取手機上的相機拍攝照片或影片，注意仍然需要開放相對的權限 (Permissions.CAMERA 與 Permissions.AUDIO_RECORDING)。

<https://docs.expo.io/versions/latest/sdk/camera/>

用巢狀結構來作背景圖片

Image 元件實際上是可作為巢狀結構的外層。

```
return (
  <Image source={...}>
    <Text>Inside</Text>
  </Image>
);
```

Expo 中另外提供了 ImageBackground 元件，可利用圖片作為背景使用。

```
return (
  <ImageBackground source={...} style={{width: '100%', height: '100%'}}>
    <Text>Inside</Text>
  </ImageBackground>
);
```

resizeMode

resizeMode 控制了圖片該如何因應外部寬度的改變。共有 5 種值可以設置。

```
enum('cover', 'contain', 'stretch', 'repeat', 'center')
```

範例:

其他小抄

如何使用圓形外角的圖片

因為 borderRadius 無法直接設置為 borderRadius: 50%。因此你需要用圖片的高與寬(width/height)來計算，設置 borderRadius 為剛好 1/2 的高與寬(width/height)。當然，圖片原本必需是個正方形，也就是高與寬要相同。

例如 100x100 的圖片:

```
// component
<Image style={styles.image} source={{ uri: 'http://placeholder.it/100x100' }} />
// styles
var styles = StyleSheet.create({
  image: {
    height: 100,
    borderRadius: 50,
    width: 100,
  },
})
```

Image 元件中可以有子元件

RN 中的 <Image/> 元件與 HTML 中的 元素的設計並不相同，Image 元件中可以包含有其它子元件。以下為範例:

```
<Image source={{ uri: 'http://placeholder.it/300x300' }}>
  <Image source={{ uri: 'http://placeholder.it/100x100' }} />
  <Text>This is my text</Text>
</Image>
```

圖片逐步載入或預先載入縮圖

因圖片檔案較大，受到網路連線或手機效能的影響，造成載入需要一定時間，載入時有很多種技巧，例如使用縮圖先載入，或是加入逐步載入的指示等等。可參考以下的教學或使用對應的套件來達成:

- 教學: [Progressive Image Loading in React Native](#)
- 教學: [Creating an Asynchronous Loading Image Component in React Native](#)
- 套件: [react-native-image-progress](#)
- 套件: [react-native-expo-image-cache](#)

圖片測試環境使用工具

在某些時候我們需要使用弱網站模擬環境進行遠端圖片下載的測試，一般常見的工具如下:

- iOS(Xcode) only: [Network Link Conditioner](#)
- 全平台: [Fiddler](#)

App icon

以 Expo 建立的專案，App icon 為 1024x1024 的 png 檔案，預設位置在 app.json 檔案中設定:

```
"icon": "./assets/images/icon.png"
```

Splash Screen(App 開頭畫面)

以 Expo 建立的專案，Splash Screen(App 開頭畫面)的尺寸為 1242 pixels 寬 x 2436 pixels 高。圖片格式以 png 為佳。

為了達到在所有手機螢幕規格都能滿版顯示，最佳的方式是使用與圖片格式相同的背景顏色來填滿:

```
"splash": {  
  "image": "./assets/splash.png",  
  "backgroundColor": "#FEF9B0"  
}
```

其它有關於圖片的填滿方式、出現效果與相關設定，詳見以下的文件內容:

- https://docs.expo.io/versions/latest/guides/splash-screens/#__next