import Video from 'react-native-video';

// 视频示例

<Video

    source={{ uri: Weburl.handleWeburl(this.getVideoUrl()) }}          // Can be a URL or a local file.

    ref={(ref) => {

        this.player = ref

    }}

    // 缩放模式

    resizeMode='contain'

    // 是否循环播放

    repeat={setting.loopPlay == true}

    // 是否暂停

    paused={this.state.paused}

    // 是否全屏

    fullscreen={this.state.fullscreen}

    // 未播放时显示的图片

    poster={Weburl.handleWeburl(this.getImgUrl())}

    onBuffer={this.onBuffer}                // Callback when remote video is buffering

    onError={this.videoError}               // Callback when video cannot be loaded

    style={[styles.backgroundVideo]}

    // 进度事件

    onProgress={(e) => {

        // currentTime 当前时间，seekableDuration 总时间

        let progress = `${parseInt((e.currentTime / e.seekableDuration) \* 100)}%`

        this.setState({ progress: progress, progressSecond: e.currentTime });

    }}

    // 结束事件

    onEnd={() => {

        this.setState({ progress: "100%", paused: true, progressSecond: 0 });

    }}

    // 加载完成，可以播放事件

    onLoad={(e) => {

        let minute = parseInt(e.duration / 60);

        let second = parseInt(e.duration % 60);

        this.setState({ minute: minute, second: second });

        // 设置开始播放的时间

        this.player.seek(this.state.progressSecond);

    }}

/>

// 音频示例

<Video

    audioOnly

    source={{ uri: this.getUrl() }}          // Can be a URL or a local file.

    ref={(ref) => {

        this.player = ref

    }}

    // 是否暂停

    paused={this.state.paused}

    onBuffer={this.onBuffer}                // Callback when remote video is buffering

    onError={this.videoError}               // Callback when video cannot be loaded

    style={[styles.backgroundVideo, { height: this.videoHeight }]}

    onProgress={(e) => {

        let progress = `${parseInt((e.currentTime / e.seekableDuration) \* 100)}%`

        this.setState({ progress: progress });

    }}

    onEnd={() => {

        this.setState({ progress: "100%", paused: true });

    }}

    onLoad={(e) => {

        let minute = parseInt(e.duration / 60);

        let second = parseInt(e.duration % 60);

        this.setState({ minute: minute, second: second })

    }}

/>