

Pannellum

- Hogar (<https://pannellum.org/>)
- Funciones (<https://pannellum.org/features/>)
- Descargar (<https://pannellum.org/download/>)
- Documentación (<https://pannellum.org/documentation/overview/>)

Descripción general y tutorial (https://pannellum.org/documentation/overview/)
Ejemplos (https://pannellum.org/documentation/examples/simple-example/)
Referenciade configuración (https://pannellum.org/documentation/reference/)
API (https://pannellum.org/documentation/api/)

Referencia

Parámetros de configuración de URL

Los parámetros de URL se utilizan para configurar Pannellum. Si una imagen se está utilizando sin algunas de las características más avanzadas de Pannellum, Pannellum puede configurarse : de URL; de lo contrario, un archivo de configuración JSON necesita para ser utilizado con el archivo `.equirectangular` `config`

config

Especifica la dirección URL de un archivo de configuración JSON.

Otros parámetros

Un subconjunto de las opciones del archivo de configuración JSON se puede utilizar como parámetros de URL. Estos incluyen `,``,``,` y `.panorama` `config` `author` `title` `hfov` `minHfov` `maxHfov` `pitch` `minPitch` `maxPitch` `yaw` `minYaw` `maxYaw` `haov` `vaov` `vOffset` `autoLoad` `autoRotate` `firstScene` `ignoreGPanoXMP`

Opciones del archivo de configuración JSON

Opciones generales

type (cadena)

Esto especifica el tipo de panorama. Puede ser `,` o `.` El valor predeterminado es `.equirectangular` `cubemap` `multires` `equirectangular`

title (cadena)

Si se establece, el valor se muestra como el título del panorama. Si no se desea ningún título, No establezca este parámetro.

author (cadena)

Si se establece, el valor se muestra como el autor del panorama. Si no hay autor deseado, no establezca este parámetro.

authorURL (cadena)

Si se establece, el texto del autor que se muestra tiene un hipervínculo a esta dirección URL. Si no hay una URL de autor es lo que desea, no establezca este parámetro. El parámetro también d para que este parámetro tenga un efecto. `author`

strings (diccionario)

Permite cambiar / traducir las cadenas orientadas al usuario. Consulte la definición en para obtener más detalles. `defaultConfig.strings` `pannellum.js`

basePath (cadena)

Esto especifica una ruta base desde la que cargar las imágenes.

autoLoad (booleano)

Cuando se establece en `,` la panorámica se cargará automáticamente. Cuando `,` el El usuario debe hacer clic en el botón Cargar para cargar el panorama. El valor predeterminado es `. true` `false`

autoRotate (número)

Al establecer este parámetro, la panorámica gira automáticamente cuando se carga. El valor especifica la velocidad de rotación en grados por segundo. Positivo es en sentido contrario a las a negativo en el sentido de las agujas del reloj.

autoRotateInactivityDelay (número)

Establece el retardo, en milisegundos, para que comience a rotar automáticamente la panorámica después de que cese la actividad del usuario. Este parámetro solo tiene efecto si se establec de iniciar la rotación, se realiza un desplazamiento panorámico del visor al tono inicial. autoRotate

autoRotateStopDelay (número)

Establece el retardo, en milisegundos, para que deje de girar automáticamente la panorámica después de que se cargue. Este parámetro solo tiene efecto si se establece el parámetro. autoRo

fallback (cadena)

Si se establece, el valor se utiliza como URL para un visor alternativo en caso de que Pannellum sea no es compatible con el dispositivo del usuario. El usuario tendrá la opción de hacer clic en esta URL si Pannellum no funciona.

orientationOnByDefault (booleano)

Si se establece en , el control de orientación del dispositivo se utilizará cuando la panorámica sea cargado, si el dispositivo lo admite. Si es false, el control de orientación del dispositivo neces pulsando un botón. El valor predeterminado es . true false

showZoomCtrl (booleano)

Si se establece en , no se mostrarán los controles de zoom. El valor predeterminado es . false true

keyboardZoom (booleano)

Si se establece en , se desactivará el zoom con el teclado. El valor predeterminado es . false true

mouseZoom (booleano o cadena)

Si se establece en , se desactivará el zoom con la rueda del ratón. El valor predeterminado es . También se puede establecer en , en cuyo caso solo se habilita cuando el El visor es de pantalla completa. false true fullscreenonly

draggable (booleano)

Si se establece en , el arrastre del ratón y táctil está desactivado. El valor predeterminado es . false true

friction (número)

Controla la "fricción" que ralentiza el movimiento del espectador después de arrastrarlo y liberado. Los valores más altos significan que el movimiento se detiene más rápido. Debe establecer predeterminado es 0,15.

disableKeyboardCtrl (booleano)

Si se establece en , los controles del teclado están desactivados. El valor predeterminado es . true false

showFullscreenCtrl (booleano)

Si se establece en , no se mostrará el control de pantalla completa. El valor predeterminado es . El botón de pantalla completa solo se mostrará si el navegador es compatible con la API de par completa. false true

showControls (booleano)

Si se establece en , no se muestra ningún control. El valor predeterminado es . false true

touchPanSpeedCoeffFactor (número)

Ajusta la velocidad de panorámica desde las entradas táctiles. El valor predeterminado es . 1

yaw (número)

Establece la posición de guiñada inicial de la panorámica en grados. El valor predeterminado es . 0

pitch (número)

Establece la posición de paso inicial de la panorámica en grados. El valor predeterminado es . 0

hfov (número)

Define el campo de visión horizontal inicial de la panorámica en grados. El valor predeterminado es . 100

minYaw y (número) maxYaw

Establece la guiñada mínima/máxima en la que puede estar el borde del visor, en grados. El valor predeterminado es / , es decir, sin límite. `-180 180`

minPitch y (número) maxPitch

Establece el tono mínimo/máximo en el que puede estar el borde del visor, en grados. El valor predeterminado es , por lo que el centro del visor puede alcanzar / . `undefined -90 90`

minHfov y (número) maxHfov

Establece el campo de visión horizontal mínimo/máximo, en grados, que el viewer se puede establecer en. El valor predeterminado es / . A menos que el parámetro se establezca en , el parám las panorámicas. `50 120 multiResMinHfov true minHfov multires`

multiResMinHfov (booleano)

Cuando se establece en , el parámetro se ignora para las panorámicas; Se utiliza un campo de visión horizontal mínimo calculado automáticamente en lugar de. El valor predeterminado es `. false minHfov multires false`

compass (booleano)

Si , se muestra una brújula. Normalmente, el valor predeterminado es ; El valor predeterminado es si la información de encabezado está presente en los metadatos XMP de Photo Sphere. `true`

northOffset (número)

Establezca el desplazamiento, en grados, del centro de la panorámica desde el norte. Como este afecta a la brújula, solo tiene efecto si se establece en `. compass true`

preview (cadena)

Especifica una dirección URL para que se muestre una imagen de vista previa antes de que se cargue la panorámica.

previewTitle (cadena)

Especifica el título que se mostrará mientras se muestra el botón de carga.

previewAuthor (cadena)

Especifica el autor que se mostrará mientras se muestra el botón de carga.

horizonPitch y (número) horizonRoll

Especifica el cabeceo / balanceo del horizonte de la imagen, en grados (para corregir panoramas no nivelados).

animationTimingFunction (función) [Solo API]

Esto especifica una función de temporización que se utilizará para animar movimientos como cuando se llama al método. La función de temporización predeterminada es . Si se especifica una personalizada, debe tomar un número [0, 1] como su único argumento y devuelve un número [0, 1]. `lookAt easeInOutQuad`

escapeHTML (booleano)

Cuando es true, HTML se escapa de las cadenas de configuración para ayudar a mitigar posibles Ataques DOM XSS. Esto es siempre cuando se utiliza el visor independiente, ya que la configur: a través de la URL; El valor predeterminado es pero puede ser establézcalo en cuando se usa la API. `true false true`

crossOrigin (cadena)

Esto especifica el tipo de solicitud CORS utilizada y se puede establecer en o . El valor predeterminado es `. anonymous use-credentials anonymous`

hotSpots (objeto)

Esto especifica un diccionario de puntos calientes que pueden ser enlaces a otras escenas, información, o enlaces externos. Cada elemento de la matriz tiene las siguientes propiedades.

pitch (número)

Especifica la parte de inclinación de la ubicación del punto caliente, en grados.

yaw (número)

Especifica la parte de guiñada de la ubicación del punto caliente, en grados.

type (cadena)

Especifica el tipo de zona activa. Puede ser para enlaces de escenas o para puntos calientes de información. Se requiere un archivo de configuración de recorrido para los puntos calientes. `scene`

text (cadena)

Esto especifica el texto que se muestra cuando el usuario pasa el cursor sobre el mancha.

URL (cadena)

Si se especifica para una zona activa, la zona activa se vincula a la dirección URL especificada. No aplicable para puntos calientes. `info scene`

attributes (dictado)

Especifica los atributos de enlace de la URL. Si no se establece, el atributo se establece en `target`, para abrir el enlace en una nueva pestaña para evitar que se abra en el marco / página del visor. `target`

`sceneId` (cadena)

Specifies the ID of the scene to link to for hot spots. Not applicable for hot spots. `scene` `info`

`targetPitch` (number)

Specifies the pitch of the target scene, in degrees. Can also be set to `same`, which uses the current pitch of the current scene as the initial pitch of the target scene. `same`

`targetYaw` (number)

Specifies the yaw of the target scene, in degrees. Can also be set to `same` or `direct`. These settings use the current yaw of the current scene as the initial yaw of the target scene; uses the current yaw direct account the values of both scenes to maintain the same direction with regard to north. `same` `sameAzimuth` `same` `sameAzimuth` `northOffset`

`targetHfov` (number)

Specifies the HFOV of the target scene, in degrees. Can also be set to `same`, which uses the current HFOV of the current scene as the initial HFOV of the target scene. `same`

`id`

Specifies hot spot ID, for use with API's function. `removeHotSpot`

`cssClass` (string)

If specified, string is used as the CSS class for the hot spot instead of the default CSS classes.

`createTooltipFunc` (function) and (object) `createTooltipArgs`

If `createTooltipFunc` is specified, this function is used to create the hot spot tooltip DOM instead of the default function. The contents of `createTooltipArgs` are passed to the function as arguments. `createTooltipFunc` `createToolt`

`clickHandlerFunc` (function) and (object) `clickHandlerArgs`

If `clickHandlerFunc` is specified, this function is added as an event handler for the hot spot's event. The event object and the contents of `clickHandlerArgs` are passed to the function as arguments. `clickHandlerFunc` `click` `click`

`scale` (boolean)

When `scale` is `true`, the hot spot is scaled to match changes in the field of view, relative to the initial field of view. Note that this does not account for changes in local image scale that occur due to distortions in the viewport. Defaults to `true`. `true` `false`

`hotSpotDebug` (boolean)

When `hotSpotDebug` is `true`, the mouse pointer's pitch and yaw are logged to the console when the mouse button is clicked. Defaults to `false`. `true` `false`

`sceneFadeDuration` (number)

Specifies the fade duration, in milliseconds, when transitioning between scenes. Not defined by default. Only applicable for tours. Only works with WebGL renderer.

`capturedKeyNumbers` (array)

Specifies the key numbers that are captured in key events. Defaults to the standard keys that are used by the viewer.

`backgroundColor` ([number, number, number])

Specifies an array containing RGB values [0, 1] that sets the background color for areas where no image data is available. Defaults to (black). For partial panoramas this applies to areas past the defined rectangle. For `equi` and (including fallback) panoramas this applies to areas corresponding to missing tiles or faces. `[0, 0, 0]` `equi` `multires` `cubemap`

`avoidShowingBackground` (boolean)

If set to `true`, prevent displaying out-of-range areas of a partial panorama by constraining the yaw and the field-of-view. Even at the corners and edges of the canvas only areas actually belonging to the panorama within `[minYaw, maxYaw]` and `[minPitch, maxPitch]` are shown, thus setting the option is not needed if this option is set. Defaults to `false`. `true` `minYaw` `maxYaw` `minPitch` `maxPitch` `backgroundColor` `false`

equi

`panorama` (string)

Sets the URL to the equi

`haov` (number)

Sets the panorama's horizontal angle of view, in degrees. Defaults to `360`. This is used if the equi

`vaov` (number)

Sets the panorama's vertical angle of view, in degrees. Defaults to `180`. This is used if the equi

`voffset` (number)

Sets the vertical offset of the center of the equi

`ignoreGPanoXMP` (boolean)

If set to `true`, any embedded Photo Sphere XMP data will be ignored; else, said data will override any existing settings. Defaults to `false`. `true` `false`

cubemap specific options

cubeMap

This is an array of URLs for the six cube faces in the order front, right, back, left, up, down. These are relative to `location` if it is set, else they are relative to the location of `viewer.html`. Absolute URLs can also be used. Images may be specified by giving instead of a URL. `basePath pannellum.htm null`

multires specific options

multiRes

This contains information about the multiresolution panorama in sub-keys.

basePath (string)

This is the base path of the URLs for the multiresolution tiles. It is relative to the regular option if it is defined, else it is relative to the location of `viewer.html`. An absolute URL can also be used. `basePath`

path (string)

This is a format string for the location of the multiresolution tiles, relative to `basePath`, which is relative to `location`. Format parameters are for the zoom level, for the cube face, for the x index, and for the y index. `multiRes.basePath basePath %l %s %x %y .extension`

fallbackPath (string)

This is a format string for the location of the fallback tiles for the CSS 3D transform-based renderer if the WebGL renderer is not supported, relative to `basePath`, which is relative to `location`. The only format parameter is the cube face. For each face, `multiRes.basePath basePath %s .extension`

extension (string)

Specifies the tiles' file extension. Do not include the `.`. `extension`

tileResolution (number)

This specifies the size in pixels of each image tile. `tileResolution`

maxLevel (number)

This specifies the maximum zoom level. `maxLevel`

cubeResolution (number)

This specifies the size in pixels of the full resolution cube faces the image tiles were created from. `cubeResolution`

Dynamic content specific options

Currently, only equirectangular dynamic content is supported.

dynamic (boolean)

The panorama source is considered dynamic when this is set to `true`. Defaults to `false`. This should be set to `true` for video. `dynamic true false true`

dynamicUpdate (boolean)

For dynamic content, viewer will start automatically updating when set to `true`. Defaults to `false`. If the updates are controlled via the `setUpdate` method, as with the Video.js plugin, this should be set to `false`. `dynamicUpdate true false setUpdate false`

Additional information for tour configuration files

A tour configuration file contains two top level properties, `default` and `scenes`. The `default` property contains options that are used for each scene, but options specified for individual scenes override these options. The `scenes` property is an array of scene objects, each required to have a property that contains the scene ID for the first scene to be displayed. The `scenes` property contains a dictionary of scenes, specified by scene IDs. The values assigned to these IDs are the scene objects. `scene.default scenes default default firstScene scenes`

API Events

load

Fired when a panorama finishes loading.

scenechange

Fired when a scene change is initiated. An event will be fired when the new scene finishes loading. Passes scene ID string to handler. `load`

fullscreenchange

Fired when browser fullscreen status changed. Passes status boolean to handler.

zoomchange

Fired when scene hfov update. Passes new HFOV value to handler.

scenechange fadedone

If a scene transition fade interval is specified, this event is fired when the fading is completed after changing scenes.

animatefinished

Fired when any movements / animations finish, i.e. when the renderer stops rendering new frames. Passes final pitch, yaw, and HFOV values to handler.

error

Fired when an error occurred. The error message string is passed to the event listener.

errorcleared

Fired when an error is cleared.

mousedown

Fired when the mouse button is pressed. Passes to handler. *MouseEvent*

mouseup

Fired when the mouse button is released. Passes to handler. *MouseEvent*

touchstart

Fired when a touch starts. Passes to handler. *TouchEvent*

touchend

Fired when a touch ends. Passes to handler. *TouchEvent*

Referencia de configuración

Parámetros de configuración de URL (<https://pannillum.org/documentation/reference/#url-configuration-parameters>)

Opciones del archivo de configuración JSON (<https://pannillum.org/documentation/reference/#json-configuration-file-options>)

Eventos de API (<https://pannillum.org/documentation/reference/#api-events>)

Pannillum is developed on [GitHub](https://github.com/mpetroff/pannillum) (<https://github.com/mpetroff/pannillum>) by [Matthew Petroff](https://mpetroff.net/) (<https://mpetroff.net/>).
CDN via [KeyCDN](https://www.keycdn.com/) (<https://www.keycdn.com/>).