

jQuery **Chop Slider 3** *by iDangero.us*

Usage Documentation

For Version 3.4

By Vladimir Kharlampidi
@nolimits4web
The iDangero.us
<http://www.idangero.us/>

Chop Slider 3
@idangerous
<http://www.idangero.us/cs/>

Thank you for purchasing the Chop Slider! If you have any questions that are beyond the scope of this help file, please feel free to contact us via our support ticket form [here](#).

Table of Contents

1. Chop Slider 3 Integration	3
1.1. Slider HTML.....	3
1.2. Slider CSS.....	4
1.3. Slider Initialization (Java Script)	4
1.4 Video Slides	12
2. Effects	13
2.1 Effects Groups	13
2.2 Effects Usage	16
2.3 Using Random Effects.....	16
2.4 Lock effects on browser support	17
3. Features	18
3.1 Navigation	18
3.2 Pagination	19
3.3 Captions	19
3.4 Ambient Light.....	21
3.5 Chop Slider Touch	21
3.6 Callback Functions.....	22
3.7 Responsive Chop Slider.....	23
3.8 Fullscreen Gallery	24
4. Skins	25
5. API.....	25

1. Chop Slider 3 Integration

1.1. Slider HTML

It is very easy to integrate Chop Slider 3 into your page. So let's first of all look at the HTML code we need to use:

First of all you need to copy "cs3" folder to your website's root folder.

Now you need to attach the jQuery library, Chop Slider script and CSS to your document. Add the following code to the HEAD section:

```
<link rel="stylesheet" href="pathToCS3/cs3/idangerous.chopslider-3.0.css"/>
<script src="pathTojQuery/jquery-1.6.1.min.js"></script>
<script src="pathToCS3/cs3/idangerous.chopslider-3.0.min.js"></script>
```

That is all, now we need to add some HTML. Insert the following code inside of BODY:

```
<div class="cs3">
  <!-- First Slide -->
  <div class="cs3-slide"></div>
  <!-- Second Slide -->
  <div class="cs3-slide"></div>
  <!-- Third Slide with link -->
  <div class="cs3-slide"><a href="http://google.com"></a></div>

  <!--Navigation Arrows -->
  <a class="cs3-slide-next" href="#"></a>
  <a class="cs3-slide-prev" href="#"></a>

  <!--Pagination-->
  <div class="cs3-pagination"></div>

  <!--Fullscreen gallery trigger -->
  <div class="cs3-fs-trigger"></div>

  <!--Captions-->
  <div class="cs3-captions">
    <div class="cs3-caption">
      <!-- Caption content of first slide -->
    </div>
    <div class="cs3-caption">
      <!-- Caption content of second slide -->
    </div>
    <div class="cs3-caption">
      <!-- Caption content of third slide -->
    </div>
  </div>
</div>
```

At the example above we used slider with three slides / images. As you can see the code is pretty simple.

This maximum of HTML code for use with navigation, pagination and captions.

Note that following classes name are reserved:

cs3 , cs3-slide, cs3-captions, cs3-caption

Also note that you can use any additional elements which you may need for styling inside of "cs3" div.

1.2. Slider CSS

Now we need to set slider's width and height with CSS. Add somewhere in your CSS files the following CSS rule:

```
.cs3 {
  width:600px;
  height:300px;
}
```

Slider's width and height should be the same as the dimension of images you use.

If you use custom navigation and pagination you will also need the following styles according to the HTML example above:

```
/* For next "arrow" */
.cs3-slide-next {
  ...
}
/* For previous "arrow" */
.cs3-slide-prev {
  ...
}
/* For pagination container */
.cs3-pagination {
  ...
}
/* For single pagination button */
.cs3-pagination-switch {
  ...
}
/* For active pagination button */
.cs3-active-switch {
  ...
}
```

1.3. Slider Initialization (Java Script)

We need to initialize the Chop Slider after the document is ready (is loaded). Use the following formatting inside of <script> tag (if you use it inside of your document file)

```
jQuery(function(){
  $(".cs3").cs3(parameters)
})
```

As you can see it was pretty simple! Now let's look at the list of available parameters:

Parameter	Type of variable	Example	Required	Default	Description
Path To CS3 folder with assets					
pathToCS3	<i>string</i>	/cs3/	optional	<code>parsed path to CS3's CSS file</code>	Path to CS3 folder with slider assets relatively to the document, or absolute URL to this folder. With trailing slash
Effects					
effects	<i>string</i>	"cube, brush"	optional	<code>random-flat'</code>	String with list of selected effects separated with comma
effectsGroupLock	<i>object</i>		optional	-	see "Effects" chapter
Ambient Light					
ambilight	<i>object</i>	70	optional	-	Object with ambilight parameters. See below.
Ambient Light Parameters Object					
enabled	<i>boolean</i>	<code>true</code>	optional	<code>false</code>	set to true if you are going to enable Ambient Light
size	<i>number</i>	70	optional	50	It will be equal to the glow size in px with each side
colorIndex	<i>number</i>	2.5	optional	1.5	Color multiplier, larger number - more lighter color
fadeIndex	<i>number</i>	1	optional	1.3	Fade multiplier, larger number - stronger fade

Parameter	Type of variable	Example	Required	Default	Description
Captions					
captions	<i>object</i>		optional	–	Object with captions parameters. See below.
Captions Parameters Object					
enabled	<i>boolean</i>	<code>true</code>	optional	<code>false</code>	set to true if you are going to use Captions
duration	<i>number</i>	<code>500</code>	optional	<code>300</code>	Duration of captions animation in ms
type	<i>string</i>	<code>"vertical"</code>	optional	<code>"horizontal"</code>	Direction of captions animation. "vertical" or "horizontal"
multi	<i>boolean</i>	<code>false</code>	optional	<code>true</code>	Set to true if you are want to enable multi Captions
multiDelay	<i>number</i>	<code>300</code>	optional	<code>100</code>	Delay between different elements in multi caption in ms
Navigation					
navigation	<i>object</i>		optional	–	Object with navigation parameters. See below.
Navigation Parameters Object					
next	<i>string</i>	<code>".cs3-slide-next"</code>	required	–	CSS selectro of element that will be used as a next navigation arrow

Parameter	Type of variable	Example	Required	Default	Description
prev	<i>string</i>	<code>".cs3-slide-prev"</code>	required	–	CSS selectro of element that will be used as a previous navigation arrow
hideOnStart	<i>boolean</i>	<code>true</code>	optional	<code>false</code>	Set to true and navigation arrows will be hidden on transition start
showOnlyOnHover	<i>boolean</i>	<code>true</code>	optional	<code>false</code>	If true then navigation will be hidden by default and will be shown only when you hover the slider.
Gallery					
gallery	<i>object</i>		optional	–	Object with gallery parameters. See below.
Gallery Parameters Object					
enabled	<i>string</i>	<code>".cs3-slide-next"</code>	required	–	CSS selector of element that will be used as a next navigation arrow
showCaptions	<i>boolean</i>	<code>true</code>	required	<code>false</code>	Set to true if you want to show captions in fullscreen gallery mode
trigger	<i>string</i>	<code>'.cs3-fs-trigger'</code>	required	–	CSS selector of element that will be used as a trigger to launch the fullscreen gallery

Parameter	Type of variable	Example	Required	Default	Description
hideOnStart	<i>boolean</i>	<code>true</code>	optional	<code>false</code>	Set to true and Gallery trigger will be hidden on transition start
showOnlyOnHover	<i>boolean</i>	<code>true</code>	optional	<code>false</code>	If true then navigation will be hidden by default and will be shown only when you hover the slider.
Pagination					
pagination	<i>object</i>		optional	–	Object with pagination parameters. See below.
Pagination Parameters Object					
container	<i>string</i>	<code>".cs3-pagination"</code>	required	–	CSS selector of element that will be used as a container with pagination buttons
hideOnStart	<i>boolean</i>	<code>true</code>	optional	<code>false</code>	Set to true and pagination will be hidden on transition start
showOnlyOnHover	<i>boolean</i>	<code>true</code>	optional	<code>false</code>	If true then pagination will be hidden by default and will be shown only when you hover the slider.
Chop Slider Touch					
touch	<i>object</i>		optional	–	Object with Chop Slider Touch parameters. See below.

Parameter	Type of variable	Example	Required	Default	Description
Chop Slider Touch Parameters Object					
enabled	<i>boolean</i>	<code>true</code>	optional	<code>false</code>	Set to true if you want to enable Chop Slider Touch plugin
effect	<i>string</i>	<code>"cube"</code> or <code>"flip-m"</code> or <code>"flip-s"</code> or <code>"slide-m"</code> or <code>"slide-s"</code>	optional	<code>"slide-s"</code>	One of 5 available touch effects
Autoplay					
autoplay	<i>object</i>		optional	<code>-</code>	Object with autoplay parameters. See below.
Auoplay Parameters Object					
enabled	<i>boolean</i>	<code>true</code>	optional	<code>false</code>	Set to true if you want to enable Autoplay
delay	<i>number</i>	<code>3000</code>	false	<code>5000</code>	Autoplay delay in ms.
disableOnInteraction	<i>boolean</i>	<code>false</code>	optional	<code>true</code>	If true, then autoplay will be disabled after you click on navigation arrow or pagination button
Preloader					
preloader	<i>boolean</i>	<code>false</code>	optional	<code>true</code>	Set to false if you want to disable integrated images preloader

Parameter	Type of variable	Example	Required	Default	Description
preloadOnlyFirst	<i>boolean</i>	<code>true</code>	optional	<code>false</code>	If set to true then preloader will preload only first slide. Useful if you have a lot of slides.
Responsive					
responsive	<i>boolean</i>	<code>true</code>	optional	<code>false</code>	Set to true if you are going to use Chop Slider in responsive layouts
Callbacks					
callbacks	<i>object</i>		optional	–	Object with callback functions. See below
Callback functions					
onInit	<i>function</i>	<code>function() {alert("Hello")}</code>	optional	–	This function will be executed after slider preload all images and its initialization
onTransitionStart	<i>function</i>	<code>function() {alert("Hello")}</code>	optional	–	This function will be executed in the beginning of transition
onTransitionEnd	<i>function</i>	<code>function() {alert("Hello")}</code>	optional	–	This function will be executed right after transition
onTouchStart	<i>function</i>	<code>function() {alert("Hello")}</code>	optional	–	This function works on touch devices with enabled Chop Slider Touch.

Parameter	Type of variable	Example	Required	Default	Description
onTouchMove	<i>function</i>	<code>function() {alert("Hello")}</code>	optional	–	This function works on touch devices with enabled Chop Slider Touch.
onTouchEnd	<i>function</i>	<code>function() {alert("Hello")}</code>	optional	–	This function works on touch devices with enabled Chop Slider Touch.

Example with parameters

Ok, let's look at the Slider initialization example again but with all possible parameters:

```
$(document).ready(function(){
    $('.cs3').cs3({

        /* Effects */
        effects : 'typewriter',
        effectsGroupLock : {
            support3d : 'threeD',
            support2d : 'canvas',
            supportCanvasNoCSS3 : 'canvas'
        },

        /* Navigation */
        navigation : {
            next : '.cs3-slide-next',
            prev : '.cs3-slide-prev',
            hideOnStart : true,
            showOnlyOnHover : false
        },

        /* Pagination */
        pagination : {
            container : '.cs3-pagination',
            hideOnStart : false,
            showOnlyOnHover : false
        },

        /* Gallery */
        gallery : {
            enabled : true,
            trigger : '.cs3-fs-trigger',
            showCaptions : true,
            hideOnStart : true,
            showOnlyOnHover : false
        }
    });
});
```

```

/* Chop Slider Touch */
touch : {
    enabled : true,
    effect : 'slide-s'
},

/* Ambient Light */
ambilight : {
    enabled : true,
    size : 100
},

/* Captions */
captions : {
    enabled : false,
    duration : 300,
    type : 'horizontal',
    multi : true,
    multiDelay : 100
},

/* Autoplay */
autoplay : {
    enabled : true,
    delay : 5000,
    disableOnInteraction : true
},

/* Responsive */
responsive : true,

/* Preloader */
preloader : true,

/* Callbacks */
callbacks : {
    onInit : function(){ alert(' Slider initialized ') },
    onTransitionStart : function(){ alert('Transition has began') },
    onTransitionEnd : function(){ alert('Transition complete') },
    onTouchStart : function() { alert('You just touch me') },
    onTouchMove : function(){ alert('You keep touching me') },
    onTouchEnd : function(){ alert('You stop touch me ') }
}

});
});

```

That was example with almost all possible parameters. Of course, **there is no sense to specify parameters that you do not need.**

1.4 Video Slides

From version 3.4 Chop Slider supports video slides (with Youtube and Vimeo iframes).

Youtube Slide

To add an Youtube slide you need to use the following HTML layout for your slide:

```
...other slides goes here
<div class="cs3-slide cs3-video-slide">
  <div class="cs3-video">
    <iframe src="http://www.youtube.com/embed/CK-a_fX3_AE" frameborder="0"
allowfullscreen></iframe>
  </div>
</div>
... another slides there
```

As you may see it is pretty easy all you need is:

- add additional **cs3-video-slide** to your slide
- wrap Youtube iframe with div with a **cs3-video** class

Vimeo Slide

To add Vimeo slide you need to use the same technic that with Youtube slide but with Vimeo iframe instead:

```
...other slides goes here
<div class="cs3-slide cs3-video-slide">
  <div class="cs3-video">
    <iframe src="http://player.vimeo.com/video/52861634?api=1" frameborder="0"
webkitAllowFullScreen mozallowfullscreen allowFullScreen></iframe>
  </div>
</div>
... another slides there
```

As you see it is almost the same, but in Vimeo iframe you also must to add «**api=1**» to the iframe URL. It is required for proper work of Vimeo API

2. Effects

Chop Slider 3 comes with a lot of unique and beautiful effects. All effects are divided by groups:

2.1 Effects Groups

Flat Effects

Simple effects which are made for compatibility with older browsers. These effects work in all browsers (desktop and mobile), even in Internet Explorer 6.

Effect Name	Effect JS Alias
Bricks	<code>bricks</code>
Horizontal Blinds	<code>blinds_h</code>
Vertical Blinds	<code>blinds_v</code>
Zip	<code>zip</code>
Fade	<code>fade</code>

2D Effects

2D effects work in browsers with support of CSS3 transforms and transitions. Supported by all major browsers like Opera, Firefox, Chrome, Safari, IE10+, Mobile Safari (iOS), Android browsers.

Effect Name	Effect JS Alias
Smear	<code>smear</code>
Horizontal Slide	<code>slide_h</code>
Vertical Slide	<code>slide_v</code>
Horizontal Zip	<code>zip_h</code>
Vertical Zip	<code>zip_v</code>
Gravity	<code>gravity</code>
Bulb	<code>bulb</code>
Morpher	<code>morpher</code>
Reveal	<code>reveal</code>

3D Effects

The most eye-catching effects. Supported by Chrome, Safari, Firefox 11+, IE10+, Mobile Safari (iOS), latest Android browsers.

Effect Name	Effect JS Alias
Turn	<code>turn</code>
3D Bricks	<code>bricks3d</code>
3D Tiles	<code>tiles3d</code>
Polaroid	<code>polaroid</code>
Horizontal Single-flip	<code>flip_h_single</code>
Horizontal Multi-flip	<code>flip_h_multi</code>
Vertical Single-flip	<code>flip_v_single</code>
Vertical Multi-flip	<code>flip_v_multi</code>
Random Flip	<code>flip_random</code>
Vertical Blocks 1	<code>blocks_v_1</code>

Effect Name	Effect JS Alias
Vertical Blocks 2	<code>blocks_v_2</code>
Vertical Blocks 3	<code>blocks_v_3</code>
Vertical Blocks 4	<code>blocks_v_4</code>
Vertical Blocks 5	<code>blocks_v_5</code>
Vertical Blocks 6	<code>blocks_v_6</code>
Vertical Blocks 7	<code>blocks_v_7</code>
Horizontal Blocks 1	<code>blocks_h_1</code>
Horizontal Blocks 2	<code>blocks_h_2</code>
Horizontal Blocks 3	<code>blocks_h_3</code>
Cube	<code>cube</code>
Horizontal Paper	<code>paper_h</code>
Vertical Paper	<code>paper_v</code>
Galaxy	<code>galaxy</code>
Explosion	<code>explosion</code>
Horizontal Panels	<code>panels_h</code>
Vertical Panels	<code>panels_v</code>

Canvas Effects

Brand new and amazingly impressive effects. They work in browsers with support of Canvas like Opera, Firefox, Chrome, Safari, IE9, Mobile Safari (iOS), Android browsers.

Effect Name	Effect JS Alias
Lines	<code>lines</code>
Razor	<code>razor</code>
Aquarium	<code>aquarium</code>
Color Burn	<code>burn</code>
Melt	<code>melt</code>
Rollover	<code>roll</code>

Effect Name	Effect JS Alias
Puzzles	puzzles
Diamonds	diamonds
Circles	circles
Paint Brush	brush
Typewriter	typewriter
Circle Reveal	circle_reveal

2.2 Effects Usage

Ok, but how to use the desired effects? It is very simple all you need is to pass the list of desired aliases to the "effects" parameter separated by comma. For example we like and want to use 3D "cube", 2D "smear" and canvas "puzzle" :

```
$( '.cs3' ).cs3({
    .....
    effects: 'cube, puzzle, smear'
    ....
})
```

That is all, now Chop Slider's core will run transition with randomly picked effect from specified in "effects" parameter.

As you can see it was ultra simple.

But what with browser support? Chop Slider 3 will take care about it. If some of effects are not supported by the browser, it will detect it and just will not use them.

If **non of specified effects are supported** (for example you have specified only 3d effects and visitor's browser do not support 3d effects) then Chop Slider 3 will show randomly picked "Flat" effect which works in all browsers.

2.3 Using Random Effects

Chop Slider 3 allows you to specify not only single effects but even entire groups, like in this example:

```
$( '.cs3' ).cs3({
    .....
    effects: 'random-3d, puzzle, random-2d'
    ....
})
```

In this example slider will pick random effect from all 2d effects, all 3d effects and canvas "puzzle" effect.

The following groups are possible:

```
$('.cs3').cs3({
    .....
    effects: 'random-2d, random-3d, random-flat, random-canvas'
    ....
})
```

In this case it will pick random effects from all possible effects.

Don't forget that you can mix single effects and effects group together!

2.4 Lock effects on browser support

Chop Slider 3 allows to lock effects depending on browser support. How does it works?

For example we've specified the following effects:

```
$('.cs3').cs3({
    .....
    effects: 'puzzle, random-3d, morpher, random-flat'
    ....
})
```

In this case CS3 will show random effect from canvas "puzzle" effect, all 3d effects, all flat effects and single 2d "morpher" effect.

But what if i want the following:

- i do not want to show 2d, flat and canvas effects to visitors with browser with 3d support. Let they see only beautiful 3d effects
- i do not want to show simple flat effects for visitors with support of CSS3 transforms but without support of 3D. Let they see only 2D and Canvas effects
- and i don't want to show flat effects for browsers with support of canvas and without support of CSS3 transitions (like Internet Explorer 9):

We can use one additional parameter:

```
$('.cs3').cs3({
    .....
    effects: 'puzzle, random-3d, morpher',
    effectsGroupLock : {
        support3d : 'threeD',
        support2d : 'canvas, twoD',
        supportCanvasNoCSS3 : 'canvas'
    },
    ...
})
```

Ok, what we've got here:

- **support3d**. You can specify here effects groups that will be shown in browsers with support of 3D effects. Example of such kind of browsers: FireFox 11+, Chrome, Safari, iOS, latest Android browsers

- **support2d**. You can specify here effects groups that will be shown in browsers with support of 2D effects and without support of 3D. Example of browsers: Opera, Android 2-3 browsers
- **supportCanvasNoCSS**. You can specify here effects groups that will be shown in browsers with support of Canvas effects and without support of 3D and 2D effects. Example of browsers: Internet Explorer 9

Available list of groups for usage with separate by comma:

- **threeD**
- **twoD**
- **canvas**

3. Features

Chop Slider 3 comes with lot of useful and exclusive features, let's look at them more closely:

3.1 Navigation

So to use navigation arrows, first of all we need to add HTML for them, like was in first example:

```
...
<!--Navigation Arrows -->
<a class="cs3-slide-next" href="#"></a>
<a class="cs3-slide-prev" href="#"></a>
...
```

On slider initialization we need to specify navigation parameter with object with options:

```
$('.cs3').cs3({
  ...
  /* Navigation and its options */
  navigation : {
    // CSS selector of next arrow. According to the HTML example above:
    next : '.cs3-slide-next',

    // CSS selector of previous arrow. According to the HTML example
above:
    prev : '.cs3-slide-prev',

    // If want to hide navigation arrows on transition start, set to true
    hideOnStart : true,

    // If want the navigation arrows was hidden by default and show
    // only on mouse hover set this option to true
    showOnlyOnHover : false
  }
  ...
})
```

That is all about Navigation, as you can see it is pretty simple

3.2 Pagination

If we want to use pagination, first of all we need to add HTML for it, like was in first example:

```
...
<!--Pagination-->
<div class="cs3-pagination"></div>
...
```

On slider initialization we need to specify Pagination parameter with object with options:

```
$('.cs3').cs3({
  ...
  /* Pagination and its options */
  pagination : {
    // CSS selector of container with pagination. According to the HTML
    example above:
    container : '.cs3-pagination',

    // If want to hide pagination on transition start, set to true
    hideOnStart : true,

    // If want the pagination was hidden by default and show it
    // only on mouse hover set this option to true
    showOnlyOnHover : false
  }
  ...
})
```

After slider initialization, it will create in this container the following HTML:

```
<div class="cs3-pagination">
  <div class="cs3-pagination-switch cs3-active-switch"></div>
  <div class="cs3-pagination-switch"></div>
  <div class="cs3-pagination-switch"></div>
  ...
</div>
```

It will create as many "cs3-pagination-switch" divs as number of slides in slider. Active switch will have additional "cs3-active-switch" class. You can use these CSS classes for customization.

3.3 Captions

Chop Slider 3 comes with brand new Captions which are ready to use with any HTML content inside. And now they work and look much more interesting and similar to mOover's "Fader" effect.

Let's find out how to use it.

First of all we need to add the following HTML inside of our "cs3" container with slider, like in the example from Chapter 1:

```
<!--Captions-->
<div class="cs3-captions">
  <!-- Caption of first slide -->
  <div class="cs3-caption">
    <!-- Caption content of first slide -->
  </div>
```

```

<!-- Caption of second slide -->
<div class="cs3-caption">
  <!-- Caption content of second slide -->
</div>
<!-- Caption of third slide -->
<div class="cs3-caption">
  <!-- Caption content of third slide -->
</div>
...
</div>

```

Number of "cs3-caption" divs must be the same as the number of slides in slider. If you don't want to use caption for some slide, just leave "cs3-caption" div for this slide empty:

```

<!--Captions-->
<div class="cs3-captions">
  <!-- Caption of first slide -->
  <div class="cs3-caption">
    <h3>Title 1</h3>
    <p>Description 1</p>
  </div>
  <!-- Caption of second slide -->
  <div class="cs3-caption">
    <h3>Title 2</h3>
    <p>Description 2</p>
  </div>
  <!-- We don't use caption for third slide. It should be empty -->
  <div class="cs3-caption"></div>
  ...
</div>

```

Multi Captions vs Single Captions

If you use Multi captions, than CS3 caption engine will animate **each children of "cs3-caption" div**. It will animate their "margin-left" (for horizontal type) and "margin-top" (for vertical type) properties. So in the HTML above, it will animate separately H3 and P inside of every caption.

If you use Single captions, than CS3 caption engine will animate **each "cs3-caption" div**. It will animate their "margin-left" (for horizontal type) and "margin-top" (for vertical type) properties. So in the HTML above, it will animate just "cs3-caption" container.

After HTML we need to enable captions and specify their options:

```

$( '.cs3' ).cs3({
  ...
  /* Captions and its options */
  captions : {
    // Set to true to enable captions:
    enabled : true,

    // Duration of captions animation in ms
    duration : 300,

    // "vertical" or "horizontal" animation:

```

```

        type : "vertical",

        // Set to true to enable multi captions:
        multi : true,

        // delay between multi captions:
        multiDelay : "vertical"
    }
    ...
})

```

That was pretty simple and self-explaining. Don't forget to check "captions" demo in demos/ folder from the downloaded archive.

3.4 Ambient Light

Ambient Light is the one of the biggest advantages of Chop Slider 3 and very exclusive features. It is very easy to use it. Just add "ambilight" object with its parameters:

```

$( '.cs3' ).cs3({
    ...
    ambilight : {
        /* Set to true to enable Ambient light */
        enabled: true,
        /* Size of ambient light in px */
        size: 70,

        /* Color brightness, the larger number - lighter color. Default value
is 1.5 */
        colorIndex:2,

        /* Fade multiplier. Larger number - stronger fade effect. Default
value is 1.3 */
        fadeIndex: 1
    },
    ...
})

```

That is all, now you will see awesome Ambient Light effect with 80px height from both sides (top and bottom).

Also note, that Ambient Light will not work in browsers without support of Canvas, like old Internet Explorer 6, 7 and 8.

3.5 Chop Slider Touch

Chop Slider Touch is one of the main features of Chop Slider 3.

We've combined our best mobile/touch experience and mix all the best from Swiper (for simple "Slide" effect) and S6 Slider (for "Cube" effect) with three more exclusive Flips and Multi-Slide effects in one small plugin - Chop Slider Touch which is already integrated in main core script.

Chop Slider Touch (if enabled) will automatically intercept desktop effects and enable touch mode with special touch effects if browser support touch. These effects work on all modern mobile/touch devices including iOS (all effects), Android 2.1+ (slide effects) and Android 3+ (all effects).

Let's look on how to use this awesome feature:

```
$( '.cs3' ).cs3({
    ...
    /* We need to specify "touch" parameter with its options: */
    touch : {
        // Set "enabled" to true if you want to enable CS Touch:
        enabled : true,

        // Now we have to choose one from 5 touch effects, for example "cube":
        effect : 'cube'
    }
    ...
})
```

The following effects are possible:

- "cube". 3D effect, works in mobile browsers with support of 3D
- "flip-s". Single Flip. 3D effect, works in mobile browsers with support of 3D
- "flip-m". Multi Flip. 3D effect, works in mobile browsers with support of 3D
- "slide-s". Single Slide. 2D effect, works almost in all mobile browsers
- "slide-m". Multi Slide. 2D effect, works almost in all mobile browser

At the moment you can specify only one touch effect.

If you will specify 3D effect and mobile browser of visitor will not support 3D, than CS Touch will automatically switch selected 3D effect to 2D effect. Here is how degradation work

- "cube" degrades to "slide-s"
- "flip-s" degrades to "slide-s"
- "flip-m" degrades to "slide-m"

Also note that if CS Touch is enabled, than effects specified in "effects" slider's parameter will be ignored for mobile browsers.

3.6 Callback Functions

Chop Slider 3 provides you with 6 callback functions. They are very useful if you want to use custom navigation, pagination or some interaction with environment.

If you want to use callback functions you need to add "callbacks" parameters with its options:

```
$( '.cs3' ).cs3({
    ...
```

```

/* Add callbacks */
callbacks : {
    // onInit callback will be executed after pre-loader will load all
images and slider is initialized
    onInit : function(){
        alert('Slider loaded')
    },

    // onTransitionStart callback will be executed right in the beginning
of transition
    onTransitionStart : function(){
        alert('Transition began')
    },

    // onTransitionEnd callback will be executed right after transition
effect
    onTransitionEnd : function(){
        alert('Transition completed')
    },

    /* Three other callbacks works only with and can be called only when
Chop Slider Touch is enabled */

    // onTouchStart callback will be executed when visitor will touch the
slider
    onTouchStart : function(){
        alert('You've touch the slider')
    },

    // onTouchMove callback will be executed when visitor will move finger
over the slider
    onTouchMove : function(){
        alert('Stop touching me!')
    },

    // onTouchEnd callback will be executed when visitor will release the
slider
    onTouchEnd : function(){
        alert('Thanks, you've just release me:!)')
    },

    }
    ...
})

```

3.7 Responsive Chop Slider

If you want to use Chop Slider 3 in responsive layouts, you need to enable this feature.

To make it responsive just add one parameter with "true" value:

```

$('.cs3').cs3({
    ...
    /* Enable responsive Chop Slider */
    responsive : true,

```

```
...
})
```

When it responsive it will updates own size on transition start and make necessary modifications to the animation core.

It is highly recommended to use it with CSS3 media queries like:

```
@media screen and (max-width: 980px) {
    .cs3 {
        width: 980px;
        height: 300px
    }
}
@media screen and (max-width: 680px) {
    .cs3 {
        width: 640px;
        height: 250px
    }
}
@media screen and (max-width: 320px) {
    .cs3 {
        width: 320px;
        height: 150px
    }
}
```

Do not enable "responsive" feature if you are not going to use it in responsive layouts!

3.8 Fullscreen Gallery

If we want to use fullscreen gallery, first of all we need to add HTML for it, like was in first example:

```
...
<!--Gallery-->
<div class="cs3-fs-trigger"></div>
...
```

This element will work like a button to launch the fullscreen mode.

On slider initialization we need to specify Gallery parameter with object with options:

```
$('.cs3').cs3({
    ...
    /* Gallery and its options */
    gallery : {
        // CSS selector of element that will launch fullscreen. According to
the HTML example above:
        trigger : '.cs3-fs-trigger',

        // If want to hide trigger on transition start, set to true
        hideOnStart : true,

        // If want the trigger was hidden by default and show it
        // only on mouse hover set this option to true
        showOnlyOnHover : false,

        // If we use captions and want show them in fullscreen mode:
```



```

        showCaptions : true
    }
    ...
})

```

4. Skins

In the package you will also find 10 beautiful and responsive visual skins ready to use for all occasions: white, black, metallic, wood, pink, 3d, plastic, etc.

All skins are configurable so you can use them with/without captions, pagination, navigation, etc.

You will find their HTML in appropriate demo from demos/ folder in downloaded package!

5. API

After you initialize Chop Slider there is a special JS object in your hands that allows you to control slider:

```
var myCS3 = $('.chopslider').cs3({...some options goes here...})
```

Now we can use "myCS3" variable/object to do some interesting thins

```
//Run transition to next slide
myCS3.slideNext()
```

```
//Run transition to previous slide
myCS3.slidePrev()
```

```
//Run transition to specific slide
myCS3.slideTo(index) //- where "index" - index number of required slide
```

```
//Immediately switch slider to other slide
myCS3.switchTo(index) //- where "index" - index number of required slide
```

```
//Get index number of currently active slide
myCS3.h.indexes().active // - returns index number of currently active slide
```

```
//Run Fullscreen gallery
myCS3.plugins.gallery.start()
```

```
//Start Autoplay
myCS3.autoplayStart()
```

```
//Stop Autoplay
myCS3.autoplayStop()
```

```
//Redefine some of init parameters
myCS3.params.autoplay.delay = 6000
```