* [Welcome to CodeIgniter](http://docs.google.com/general/welcome.html)
* [Installation Instructions](http://docs.google.com/installation/index.html)
  + [Downloading CodeIgniter](http://docs.google.com/installation/downloads.html)
  + [Installation Instructions](http://docs.google.com/installation/index.html)
  + [Upgrading From a Previous Version](http://docs.google.com/installation/upgrading.html)
  + [Troubleshooting](http://docs.google.com/installation/troubleshooting.html)
* [CodeIgniter Overview](http://docs.google.com/overview/index.html)
  + [Getting Started](http://docs.google.com/overview/getting_started.html)
  + [CodeIgniter at a Glance](http://docs.google.com/overview/at_a_glance.html)
  + [Supported Features](http://docs.google.com/overview/features.html)
  + [Application Flow Chart](http://docs.google.com/overview/appflow.html)
  + [Model-View-Controller](http://docs.google.com/overview/mvc.html)
  + [Architectural Goals](http://docs.google.com/overview/goals.html)
* [Tutorial](http://docs.google.com/tutorial/index.html)
  + [Static pages](http://docs.google.com/tutorial/static_pages.html)
  + [News section](http://docs.google.com/tutorial/news_section.html)
  + [Create news items](http://docs.google.com/tutorial/create_news_items.html)
  + [Conclusion](http://docs.google.com/tutorial/conclusion.html)
* [Contributing to CodeIgniter](http://docs.google.com/contributing/index.html)
  + [Writing CodeIgniter Documentation](http://docs.google.com/documentation/index.html)
  + [Developer’s Certificate of Origin 1.1](http://docs.google.com/DCO.html)
* [General Topics](http://docs.google.com/general/index.html)
  + [CodeIgniter URLs](http://docs.google.com/general/urls.html)
  + [Controllers](http://docs.google.com/general/controllers.html)
  + [Reserved Names](http://docs.google.com/general/reserved_names.html)
  + [Views](http://docs.google.com/general/views.html)
  + [Models](http://docs.google.com/general/models.html)
  + [Helpers](http://docs.google.com/general/helpers.html)
  + [Using CodeIgniter Libraries](http://docs.google.com/general/libraries.html)
  + [Creating Libraries](http://docs.google.com/general/creating_libraries.html)
  + [Using CodeIgniter Drivers](http://docs.google.com/general/drivers.html)
  + [Creating Drivers](http://docs.google.com/general/creating_drivers.html)
  + [Creating Core System Classes](http://docs.google.com/general/core_classes.html)
  + [Creating Ancillary Classes](http://docs.google.com/general/ancillary_classes.html)
  + [Hooks - Extending the Framework Core](http://docs.google.com/general/hooks.html)
  + [Auto-loading Resources](http://docs.google.com/general/autoloader.html)
  + [Common Functions](http://docs.google.com/general/common_functions.html)
  + [Compatibility Functions](http://docs.google.com/general/compatibility_functions.html)
  + [URI Routing](http://docs.google.com/general/routing.html)
  + [Error Handling](http://docs.google.com/general/errors.html)
  + [Caching](http://docs.google.com/general/caching.html)
  + [Profiling Your Application](http://docs.google.com/general/profiling.html)
  + [Running via the CLI](http://docs.google.com/general/cli.html)
  + [Managing your Applications](http://docs.google.com/general/managing_apps.html)
  + [Handling Multiple Environments](http://docs.google.com/general/environments.html)
  + [Alternate PHP Syntax for View Files](http://docs.google.com/general/alternative_php.html)
  + [Security](http://docs.google.com/general/security.html)
  + [PHP Style Guide](http://docs.google.com/general/styleguide.html)
* [Libraries](http://docs.google.com/index.html)
  + [Benchmarking Class](http://docs.google.com/benchmark.html)
  + [Caching Driver](http://docs.google.com/caching.html)
  + [Calendaring Class](http://docs.google.com/calendar.html)
  + [Shopping Cart Class](http://docs.google.com/cart.html)
  + [Config Class](http://docs.google.com/config.html)
  + [Email Class](http://docs.google.com/email.html)
  + [Encrypt Class](http://docs.google.com/encrypt.html)
  + [Encryption Library](http://docs.google.com/encryption.html)
  + [File Uploading Class](http://docs.google.com/file_uploading.html)
  + [Form Validation](http://docs.google.com/form_validation.html)
  + [FTP Class](http://docs.google.com/ftp.html)
  + Image Manipulation Class
  + [Input Class](http://docs.google.com/input.html)
  + [Javascript Class](http://docs.google.com/javascript.html)
  + [Language Class](http://docs.google.com/language.html)
  + [Loader Class](http://docs.google.com/loader.html)
  + [Migrations Class](http://docs.google.com/migration.html)
  + [Output Class](http://docs.google.com/output.html)
  + [Pagination Class](http://docs.google.com/pagination.html)
  + [Template Parser Class](http://docs.google.com/parser.html)
  + [Security Class](http://docs.google.com/security.html)
  + [Session Library](http://docs.google.com/sessions.html)
  + [HTML Table Class](http://docs.google.com/table.html)
  + [Trackback Class](http://docs.google.com/trackback.html)
  + [Typography Class](http://docs.google.com/typography.html)
  + [Unit Testing Class](http://docs.google.com/unit_testing.html)
  + [URI Class](http://docs.google.com/uri.html)
  + [User Agent Class](http://docs.google.com/user_agent.html)
  + [XML-RPC and XML-RPC Server Classes](http://docs.google.com/xmlrpc.html)
  + [Zip Encoding Class](http://docs.google.com/zip.html)
* [Database Reference](http://docs.google.com/database/index.html)
  + [Quick Start: Usage Examples](http://docs.google.com/database/examples.html)
  + [Database Configuration](http://docs.google.com/database/configuration.html)
  + [Connecting to a Database](http://docs.google.com/database/connecting.html)
  + [Running Queries](http://docs.google.com/database/queries.html)
  + [Generating Query Results](http://docs.google.com/database/results.html)
  + [Query Helper Functions](http://docs.google.com/database/helpers.html)
  + [Query Builder Class](http://docs.google.com/database/query_builder.html)
  + [Transactions](http://docs.google.com/database/transactions.html)
  + [Getting MetaData](http://docs.google.com/database/metadata.html)
  + [Custom Function Calls](http://docs.google.com/database/call_function.html)
  + [Query Caching](http://docs.google.com/database/caching.html)
  + [Database Manipulation with Database Forge](http://docs.google.com/database/forge.html)
  + [Database Utilities Class](http://docs.google.com/database/utilities.html)
  + [Database Driver Reference](http://docs.google.com/database/db_driver_reference.html)
* [Helpers](http://docs.google.com/helpers/index.html)
  + [Array Helper](http://docs.google.com/helpers/array_helper.html)
  + [CAPTCHA Helper](http://docs.google.com/helpers/captcha_helper.html)
  + [Cookie Helper](http://docs.google.com/helpers/cookie_helper.html)
  + [Date Helper](http://docs.google.com/helpers/date_helper.html)
  + [Directory Helper](http://docs.google.com/helpers/directory_helper.html)
  + [Download Helper](http://docs.google.com/helpers/download_helper.html)
  + [Email Helper](http://docs.google.com/helpers/email_helper.html)
  + [File Helper](http://docs.google.com/helpers/file_helper.html)
  + [Form Helper](http://docs.google.com/helpers/form_helper.html)
  + [HTML Helper](http://docs.google.com/helpers/html_helper.html)
  + [Inflector Helper](http://docs.google.com/helpers/inflector_helper.html)
  + [Language Helper](http://docs.google.com/helpers/language_helper.html)
  + [Number Helper](http://docs.google.com/helpers/number_helper.html)
  + [Path Helper](http://docs.google.com/helpers/path_helper.html)
  + [Security Helper](http://docs.google.com/helpers/security_helper.html)
  + [Smiley Helper](http://docs.google.com/helpers/smiley_helper.html)
  + [String Helper](http://docs.google.com/helpers/string_helper.html)
  + [Text Helper](http://docs.google.com/helpers/text_helper.html)
  + [Typography Helper](http://docs.google.com/helpers/typography_helper.html)
  + [URL Helper](http://docs.google.com/helpers/url_helper.html)
  + [XML Helper](http://docs.google.com/helpers/xml_helper.html)

Toggle Table of Contents

[CodeIgniter](http://docs.google.com/index.html)

* [Welcome to CodeIgniter](http://docs.google.com/general/welcome.html)
* [Installation Instructions](http://docs.google.com/installation/index.html)
  + [Downloading CodeIgniter](http://docs.google.com/installation/downloads.html)
  + [Installation Instructions](http://docs.google.com/installation/index.html)
  + [Upgrading From a Previous Version](http://docs.google.com/installation/upgrading.html)
  + [Troubleshooting](http://docs.google.com/installation/troubleshooting.html)
* [CodeIgniter Overview](http://docs.google.com/overview/index.html)
  + [Getting Started](http://docs.google.com/overview/getting_started.html)
  + [CodeIgniter at a Glance](http://docs.google.com/overview/at_a_glance.html)
  + [Supported Features](http://docs.google.com/overview/features.html)
  + [Application Flow Chart](http://docs.google.com/overview/appflow.html)
  + [Model-View-Controller](http://docs.google.com/overview/mvc.html)
  + [Architectural Goals](http://docs.google.com/overview/goals.html)
* [Tutorial](http://docs.google.com/tutorial/index.html)
  + [Static pages](http://docs.google.com/tutorial/static_pages.html)
  + [News section](http://docs.google.com/tutorial/news_section.html)
  + [Create news items](http://docs.google.com/tutorial/create_news_items.html)
  + [Conclusion](http://docs.google.com/tutorial/conclusion.html)
* [Contributing to CodeIgniter](http://docs.google.com/contributing/index.html)
  + [Writing CodeIgniter Documentation](http://docs.google.com/documentation/index.html)
  + [Developer’s Certificate of Origin 1.1](http://docs.google.com/DCO.html)
* [General Topics](http://docs.google.com/general/index.html)
  + [CodeIgniter URLs](http://docs.google.com/general/urls.html)
  + [Controllers](http://docs.google.com/general/controllers.html)
  + [Reserved Names](http://docs.google.com/general/reserved_names.html)
  + [Views](http://docs.google.com/general/views.html)
  + [Models](http://docs.google.com/general/models.html)
  + [Helpers](http://docs.google.com/general/helpers.html)
  + [Using CodeIgniter Libraries](http://docs.google.com/general/libraries.html)
  + [Creating Libraries](http://docs.google.com/general/creating_libraries.html)
  + [Using CodeIgniter Drivers](http://docs.google.com/general/drivers.html)
  + [Creating Drivers](http://docs.google.com/general/creating_drivers.html)
  + [Creating Core System Classes](http://docs.google.com/general/core_classes.html)
  + [Creating Ancillary Classes](http://docs.google.com/general/ancillary_classes.html)
  + [Hooks - Extending the Framework Core](http://docs.google.com/general/hooks.html)
  + [Auto-loading Resources](http://docs.google.com/general/autoloader.html)
  + [Common Functions](http://docs.google.com/general/common_functions.html)
  + [Compatibility Functions](http://docs.google.com/general/compatibility_functions.html)
  + [URI Routing](http://docs.google.com/general/routing.html)
  + [Error Handling](http://docs.google.com/general/errors.html)
  + [Caching](http://docs.google.com/general/caching.html)
  + [Profiling Your Application](http://docs.google.com/general/profiling.html)
  + [Running via the CLI](http://docs.google.com/general/cli.html)
  + [Managing your Applications](http://docs.google.com/general/managing_apps.html)
  + [Handling Multiple Environments](http://docs.google.com/general/environments.html)
  + [Alternate PHP Syntax for View Files](http://docs.google.com/general/alternative_php.html)
  + [Security](http://docs.google.com/general/security.html)
  + [PHP Style Guide](http://docs.google.com/general/styleguide.html)
* [Libraries](http://docs.google.com/index.html)
  + [Benchmarking Class](http://docs.google.com/benchmark.html)
  + [Caching Driver](http://docs.google.com/caching.html)
  + [Calendaring Class](http://docs.google.com/calendar.html)
  + [Shopping Cart Class](http://docs.google.com/cart.html)
  + [Config Class](http://docs.google.com/config.html)
  + [Email Class](http://docs.google.com/email.html)
  + [Encrypt Class](http://docs.google.com/encrypt.html)
  + [Encryption Library](http://docs.google.com/encryption.html)
  + [File Uploading Class](http://docs.google.com/file_uploading.html)
  + [Form Validation](http://docs.google.com/form_validation.html)
  + [FTP Class](http://docs.google.com/ftp.html)
  + Image Manipulation Class
  + [Input Class](http://docs.google.com/input.html)
  + [Javascript Class](http://docs.google.com/javascript.html)
  + [Language Class](http://docs.google.com/language.html)
  + [Loader Class](http://docs.google.com/loader.html)
  + [Migrations Class](http://docs.google.com/migration.html)
  + [Output Class](http://docs.google.com/output.html)
  + [Pagination Class](http://docs.google.com/pagination.html)
  + [Template Parser Class](http://docs.google.com/parser.html)
  + [Security Class](http://docs.google.com/security.html)
  + [Session Library](http://docs.google.com/sessions.html)
  + [HTML Table Class](http://docs.google.com/table.html)
  + [Trackback Class](http://docs.google.com/trackback.html)
  + [Typography Class](http://docs.google.com/typography.html)
  + [Unit Testing Class](http://docs.google.com/unit_testing.html)
  + [URI Class](http://docs.google.com/uri.html)
  + [User Agent Class](http://docs.google.com/user_agent.html)
  + [XML-RPC and XML-RPC Server Classes](http://docs.google.com/xmlrpc.html)
  + [Zip Encoding Class](http://docs.google.com/zip.html)
* [Database Reference](http://docs.google.com/database/index.html)
  + [Quick Start: Usage Examples](http://docs.google.com/database/examples.html)
  + [Database Configuration](http://docs.google.com/database/configuration.html)
  + [Connecting to a Database](http://docs.google.com/database/connecting.html)
  + [Running Queries](http://docs.google.com/database/queries.html)
  + [Generating Query Results](http://docs.google.com/database/results.html)
  + [Query Helper Functions](http://docs.google.com/database/helpers.html)
  + [Query Builder Class](http://docs.google.com/database/query_builder.html)
  + [Transactions](http://docs.google.com/database/transactions.html)
  + [Getting MetaData](http://docs.google.com/database/metadata.html)
  + [Custom Function Calls](http://docs.google.com/database/call_function.html)
  + [Query Caching](http://docs.google.com/database/caching.html)
  + [Database Manipulation with Database Forge](http://docs.google.com/database/forge.html)
  + [Database Utilities Class](http://docs.google.com/database/utilities.html)
  + [Database Driver Reference](http://docs.google.com/database/db_driver_reference.html)
* [Helpers](http://docs.google.com/helpers/index.html)
  + [Array Helper](http://docs.google.com/helpers/array_helper.html)
  + [CAPTCHA Helper](http://docs.google.com/helpers/captcha_helper.html)
  + [Cookie Helper](http://docs.google.com/helpers/cookie_helper.html)
  + [Date Helper](http://docs.google.com/helpers/date_helper.html)
  + [Directory Helper](http://docs.google.com/helpers/directory_helper.html)
  + [Download Helper](http://docs.google.com/helpers/download_helper.html)
  + [Email Helper](http://docs.google.com/helpers/email_helper.html)
  + [File Helper](http://docs.google.com/helpers/file_helper.html)
  + [Form Helper](http://docs.google.com/helpers/form_helper.html)
  + [HTML Helper](http://docs.google.com/helpers/html_helper.html)
  + [Inflector Helper](http://docs.google.com/helpers/inflector_helper.html)
  + [Language Helper](http://docs.google.com/helpers/language_helper.html)
  + [Number Helper](http://docs.google.com/helpers/number_helper.html)
  + [Path Helper](http://docs.google.com/helpers/path_helper.html)
  + [Security Helper](http://docs.google.com/helpers/security_helper.html)
  + [Smiley Helper](http://docs.google.com/helpers/smiley_helper.html)
  + [String Helper](http://docs.google.com/helpers/string_helper.html)
  + [Text Helper](http://docs.google.com/helpers/text_helper.html)
  + [Typography Helper](http://docs.google.com/helpers/typography_helper.html)
  + [URL Helper](http://docs.google.com/helpers/url_helper.html)
  + [XML Helper](http://docs.google.com/helpers/xml_helper.html)

[CodeIgniter](http://docs.google.com/index.html)

* [Docs](http://docs.google.com/index.html) »
* [Libraries](http://docs.google.com/index.html) »
* Image Manipulation Class

classic layout

Image Manipulation Class[**¶**](#gjdgxs)

CodeIgniter’s Image Manipulation class lets you perform the following actions:

* Image Resizing
* Thumbnail Creation
* Image Cropping
* Image Rotating
* Image Watermarking

All three major image libraries are supported: GD/GD2, NetPBM, and ImageMagick

Note

Watermarking is only available using the GD/GD2 library. In addition, even though other libraries are supported, GD is required in order for the script to calculate the image properties. The image processing, however, will be performed with the library you specify.

* [Initializing the Class](#30j0zll)
  + [Processing an Image](#1fob9te)
  + [Processing Methods](#3znysh7)
  + [Preferences](#2et92p0)
  + [Setting preferences in a config file](#tyjcwt)
* [Image Watermarking](#3dy6vkm)
  + [Two Types of Watermarking](#1t3h5sf)
  + [Watermarking an Image](#4d34og8)
  + [Watermarking Preferences](#2s8eyo1)
    - [Text Preferences](#17dp8vu)
    - [Overlay Preferences](#3rdcrjn)
* [Class Reference](#26in1rg)

[**Initializing the Class**](#lnxbz9)[**¶**](#30j0zll)

Like most other classes in CodeIgniter, the image class is initialized in your controller using the $this->load->library function:

$this->load->library('image\_lib');

Once the library is loaded it will be ready for use. The image library object you will use to call all functions is: $this->image\_lib

[**Processing an Image**](#35nkun2)[**¶**](#1fob9te)

Regardless of the type of processing you would like to perform (resizing, cropping, rotation, or watermarking), the general process is identical. You will set some preferences corresponding to the action you intend to perform, then call one of four available processing functions. For example, to create an image thumbnail you’ll do this:

$config['image\_library'] = 'gd2';  
$config['source\_image'] = '/path/to/image/mypic.jpg';  
$config['create\_thumb'] = TRUE;  
$config['maintain\_ratio'] = TRUE;  
$config['width'] = 75;  
$config['height'] = 50;  
  
$this->load->library('image\_lib', $config);  
  
$this->image\_lib->resize();

The above code tells the image\_resize function to look for an image called *mypic.jpg* located in the source\_image folder, then create a thumbnail that is 75 X 50 pixels using the GD2 image\_library. Since the maintain\_ratio option is enabled, the thumb will be as close to the target width and height as possible while preserving the original aspect ratio. The thumbnail will be called *mypic\_thumb.jpg* and located at the same level as *source\_image*.

Note

In order for the image class to be allowed to do any processing, the folder containing the image files must have write permissions.

Note

Image processing can require a considerable amount of server memory for some operations. If you are experiencing out of memory errors while processing images you may need to limit their maximum size, and/or adjust PHP memory limits.

[**Processing Methods**](#1ksv4uv)[**¶**](#3znysh7)

There are four available processing methods:

* $this->image\_lib->resize()
* $this->image\_lib->crop()
* $this->image\_lib->rotate()
* $this->image\_lib->watermark()

These methods return boolean TRUE upon success and FALSE for failure. If they fail you can retrieve the error message using this function:

echo $this->image\_lib->display\_errors();

A good practice is to use the processing function conditionally, showing an error upon failure, like this:

if ( ! $this->image\_lib->resize())  
{  
 echo $this->image\_lib->display\_errors();  
}

Note

You can optionally specify the HTML formatting to be applied to the errors, by submitting the opening/closing tags in the function, like this:

$this->image\_lib->display\_errors('<p>', '</p>');

[**Preferences**](#44sinio)[**¶**](#2et92p0)

The preferences described below allow you to tailor the image processing to suit your needs.

Note that not all preferences are available for every function. For example, the x/y axis preferences are only available for image cropping. Likewise, the width and height preferences have no effect on cropping. The “availability” column indicates which functions support a given preference.

Availability Legend:

* R - Image Resizing
* C - Image Cropping
* X - Image Rotation
* W - Image Watermarking

| Preference | Default Value | Options | Description | Availability |
| --- | --- | --- | --- | --- |
| **image\_library** | GD2 | GD, GD2, ImageMagick, NetPBM | Sets the image library to be used. | R, C, X, W |
| **library\_path** | None | None | Sets the server path to your ImageMagick or NetPBM library. If you use either of those libraries you must supply the path. | R, C, X R, C, S, W |
| **source\_image** | None | None | Sets the source image name/path. The path must be a relative or absolute server path, not a URL. |  |
| **dynamic\_output** | FALSE | TRUE/FALSE (boolean) | Determines whether the new image file should be written to disk or generated dynamically. Note: If you choose the dynamic setting, only one image can be shown at a time, and it can’t be positioned on the page. It simply outputs the raw image dynamically to your browser, along with image headers. | R, C, X, W |
| **file\_permissions** | 0644 | (integer) | File system permissions to apply on the resulting image file, writing it to the disk. WARNING: Use octal integer notation! | R, C, X, W |
| **quality** | 90% | 1 - 100% | Sets the quality of the image. The higher the quality the larger the file size. | R, C, X, W |
| **new\_image** | None | None | Sets the destination image name/path. You’ll use this preference when creating an image copy. The path must be a relative or absolute server path, not a URL. | R, C, X, W |
| **width** | None | None | Sets the width you would like the image set to. | R, C |
| **height** | None | None | Sets the height you would like the image set to. | R, C |
| **create\_thumb** | FALSE | TRUE/FALSE (boolean) | Tells the image processing function to create a thumb. | R |
| **thumb\_marker** | \_thumb | None | Specifies the thumbnail indicator. It will be inserted just before the file extension, so mypic.jpg would become mypic\_thumb.jpg | R |
| **maintain\_ratio** | TRUE | TRUE/FALSE (boolean) | Specifies whether to maintain the original aspect ratio when resizing or use hard values. | R, C |
| **master\_dim** | auto | auto, width, height | Specifies what to use as the master axis when resizing or creating thumbs. For example, let’s say you want to resize an image to 100 X 75 pixels. If the source image size does not allow perfect resizing to those dimensions, this setting determines which axis should be used as the hard value. “auto” sets the axis automatically based on whether the image is taller than wider, or vice versa. | R |
| **rotation\_angle** | None | 90, 180, 270, vrt, hor | Specifies the angle of rotation when rotating images. Note that PHP rotates counter-clockwise, so a 90 degree rotation to the right must be specified as 270. | X |
| **x\_axis** | None | None | Sets the X coordinate in pixels for image cropping. For example, a setting of 30 will crop an image 30 pixels from the left. | C |
| **y\_axis** | None | None | Sets the Y coordinate in pixels for image cropping. For example, a setting of 30 will crop an image 30 pixels from the top. | C |

[**Setting preferences in a config file**](#2jxsxqh)[**¶**](#tyjcwt)

If you prefer not to set preferences using the above method, you can instead put them into a config file. Simply create a new file called image\_lib.php, add the $config array in that file. Then save the file in *config/image\_lib.php* and it will be used automatically. You will NOT need to use the $this->image\_lib->initialize() method if you save your preferences in a config file.

[**Image Watermarking**](#z337ya)[**¶**](#3dy6vkm)

The Watermarking feature requires the GD/GD2 library.

[**Two Types of Watermarking**](#3j2qqm3)[**¶**](#1t3h5sf)

There are two types of watermarking that you can use:

* **Text**: The watermark message will be generated using text, either with a True Type font that you specify, or using the native text output that the GD library supports. If you use the True Type version your GD installation must be compiled with True Type support (most are, but not all).
* **Overlay**: The watermark message will be generated by overlaying an image (usually a transparent PNG or GIF) containing your watermark over the source image.

[**Watermarking an Image**](#1y810tw)[**¶**](#4d34og8)

Just as with the other methods (resizing, cropping, and rotating) the general process for watermarking involves setting the preferences corresponding to the action you intend to perform, then calling the watermark function. Here is an example:

$config['source\_image'] = '/path/to/image/mypic.jpg';  
$config['wm\_text'] = 'Copyright 2006 - John Doe';  
$config['wm\_type'] = 'text';  
$config['wm\_font\_path'] = './system/fonts/texb.ttf';  
$config['wm\_font\_size'] = '16';  
$config['wm\_font\_color'] = 'ffffff';  
$config['wm\_vrt\_alignment'] = 'bottom';  
$config['wm\_hor\_alignment'] = 'center';  
$config['wm\_padding'] = '20';  
  
$this->image\_lib->initialize($config);  
  
$this->image\_lib->watermark();

The above example will use a 16 pixel True Type font to create the text “Copyright 2006 - John Doe”. The watermark will be positioned at the bottom/center of the image, 20 pixels from the bottom of the image.

Note

In order for the image class to be allowed to do any processing, the image file must have “write” file permissions For example, 777.

[**Watermarking Preferences**](#4i7ojhp)[**¶**](#2s8eyo1)

This table shows the preferences that are available for both types of watermarking (text or overlay)

| Preference | Default Value | Options | Description |
| --- | --- | --- | --- |
| **wm\_type** | text | text, overlay | Sets the type of watermarking that should be used. |
| **source\_image** | None | None | Sets the source image name/path. The path must be a relative or absolute server path, not a URL. |
| **dynamic\_output** | FALSE | TRUE/FALSE (boolean) | Determines whether the new image file should be written to disk or generated dynamically. Note: If you choose the dynamic setting, only one image can be shown at a time, and it can’t be positioned on the page. It simply outputs the raw image dynamically to your browser, along with image headers. |
| **quality** | 90% | 1 - 100% | Sets the quality of the image. The higher the quality the larger the file size. |
| **wm\_padding** | None | A number | The amount of padding, set in pixels, that will be applied to the watermark to set it away from the edge of your images. |
| **wm\_vrt\_alignment** | bottom | top, middle, bottom | Sets the vertical alignment for the watermark image. |
| **wm\_hor\_alignment** | center | left, center, right | Sets the horizontal alignment for the watermark image. |
| **wm\_hor\_offset** | None | None | You may specify a horizontal offset (in pixels) to apply to the watermark position. The offset normally moves the watermark to the right, except if you have your alignment set to “right” then your offset value will move the watermark toward the left of the image. |
| **wm\_vrt\_offset** | None | None | You may specify a vertical offset (in pixels) to apply to the watermark position. The offset normally moves the watermark down, except if you have your alignment set to “bottom” then your offset value will move the watermark toward the top of the image. |

[**Text Preferences**](#2xcytpi)[**¶**](#17dp8vu)

This table shows the preferences that are available for the text type of watermarking.

| Preference | Default Value | Options | Description |
| --- | --- | --- | --- |
| **wm\_text** | None | None | The text you would like shown as the watermark. Typically this will be a copyright notice. |
| **wm\_font\_path** | None | None | The server path to the True Type Font you would like to use. If you do not use this option, the native GD font will be used. |
| **wm\_font\_size** | 16 | None | The size of the text. Note: If you are not using the True Type option above, the number is set using a range of 1 - 5. Otherwise, you can use any valid pixel size for the font you’re using. |
| **wm\_font\_color** | ffffff | None | The font color, specified in hex. Both the full 6-length (ie, 993300) and the short three character abbreviated version (ie, fff) are supported. |
| **wm\_shadow\_color** | None | None | The color of the drop shadow, specified in hex. If you leave this blank a drop shadow will not be used. Both the full 6-length (ie, 993300) and the short three character abbreviated version (ie, fff) are supported. |
| **wm\_shadow\_distance** | 3 | None | The distance (in pixels) from the font that the drop shadow should appear. |

[**Overlay Preferences**](#1ci93xb)[**¶**](#3rdcrjn)

This table shows the preferences that are available for the overlay type of watermarking.

| Preference | Default Value | Options | Description |
| --- | --- | --- | --- |
| **wm\_overlay\_path** | None | None | The server path to the image you wish to use as your watermark. Required only if you are using the overlay method. |
| **wm\_opacity** | 50 | 1 - 100 | Image opacity. You may specify the opacity (i.e. transparency) of your watermark image. This allows the watermark to be faint and not completely obscure the details from the original image behind it. A 50% opacity is typical. |
| **wm\_x\_transp** | 4 | A number | If your watermark image is a PNG or GIF image, you may specify a color on the image to be “transparent”. This setting (along with the next) will allow you to specify that color. This works by specifying the “X” and “Y” coordinate pixel (measured from the upper left) within the image that corresponds to a pixel representative of the color you want to be transparent. |
| **wm\_y\_transp** | 4 | A number | Along with the previous setting, this allows you to specify the coordinate to a pixel representative of the color you want to be transparent. |

[**Class Reference**](#3whwml4)[**¶**](#26in1rg)

*class* CI\_Image\_lib[¶](#2bn6wsx) initialize([*$props = array()*])[¶](#qsh70q)

| Parameters: | * **$props** (*array*) – Image processing preferences |
| --- | --- |
| Returns: | TRUE on success, FALSE in case of invalid settings |
| Return type: | bool |

Initializes the class for processing an image.

resize()[¶](#3as4poj)

| Returns: | TRUE on success, FALSE on failure |
| --- | --- |
| Return type: | bool |

The image resizing method lets you resize the original image, create a copy (with or without resizing), or create a thumbnail image.

For practical purposes there is no difference between creating a copy and creating a thumbnail except a thumb will have the thumbnail marker as part of the name (i.e. mypic\_thumb.jpg).

All preferences listed in the [Preferences](#1pxezwc) table are available for this method except these three: *rotation\_angle*, *x\_axis* and *y\_axis*.

**Creating a Thumbnail**

The resizing method will create a thumbnail file (and preserve the original) if you set this preference to TRUE:

$config['create\_thumb'] = TRUE;

This single preference determines whether a thumbnail is created or not.

**Creating a Copy**

The resizing method will create a copy of the image file (and preserve the original) if you set a path and/or a new filename using this preference:

$config['new\_image'] = '/path/to/new\_image.jpg';

Notes regarding this preference:

* If only the new image name is specified it will be placed in the same folder as the original
* If only the path is specified, the new image will be placed in the destination with the same name as the original.
* If both the path and image name are specified it will placed in its own destination and given the new name.

**Resizing the Original Image**

If neither of the two preferences listed above (create\_thumb, and new\_image) are used, the resizing method will instead target the original image for processing.

crop()[¶](#49x2ik5)

| Returns: | TRUE on success, FALSE on failure |
| --- | --- |
| Return type: | bool |

The cropping method works nearly identically to the resizing function except it requires that you set preferences for the X and Y axis (in pixels) specifying where to crop, like this:

$config['x\_axis'] = 100;  
$config['y\_axis'] = 40;

All preferences listed in the [Preferences](#1pxezwc) table are available for this method except these: *rotation\_angle*, *create\_thumb* and *new\_image*.

Here’s an example showing how you might crop an image:

$config['image\_library'] = 'imagemagick';  
$config['library\_path'] = '/usr/X11R6/bin/';  
$config['source\_image'] = '/path/to/image/mypic.jpg';  
$config['x\_axis'] = 100;  
$config['y\_axis'] = 60;  
  
$this->image\_lib->initialize($config);  
  
if ( ! $this->image\_lib->crop())  
{  
 echo $this->image\_lib->display\_errors();  
}

Note

Without a visual interface it is difficult to crop images, so this method is not very useful unless you intend to build such an interface. That’s exactly what we did using for the photo gallery module in ExpressionEngine, the CMS we develop. We added a JavaScript UI that lets the cropping area be selected.

rotate()[¶](#2p2csry)

| Returns: | TRUE on success, FALSE on failure |
| --- | --- |
| Return type: | bool |

The image rotation method requires that the angle of rotation be set via its preference:

$config['rotation\_angle'] = '90';

There are 5 rotation options:

1. 90 - rotates counter-clockwise by 90 degrees.
2. 180 - rotates counter-clockwise by 180 degrees.
3. 270 - rotates counter-clockwise by 270 degrees.
4. hor - flips the image horizontally.
5. vrt - flips the image vertically.

Here’s an example showing how you might rotate an image:

$config['image\_library'] = 'netpbm';  
$config['library\_path'] = '/usr/bin/';  
$config['source\_image'] = '/path/to/image/mypic.jpg';  
$config['rotation\_angle'] = 'hor';  
  
$this->image\_lib->initialize($config);  
  
if ( ! $this->image\_lib->rotate())  
{  
 echo $this->image\_lib->display\_errors();  
}

watermark()[¶](#147n2zr)

| Returns: | TRUE on success, FALSE on failure |
| --- | --- |
| Return type: | bool |

Creates a watermark over an image, please refer to the [Watermarking an Image](#3o7alnk) section for more info.

clear()[¶](#23ckvvd)

| Return type: | void |
| --- | --- |

The clear method resets all of the values used when processing an image. You will want to call this if you are processing images in a loop.

$this->image\_lib->clear();

display\_errors([*$open = '<p>*[, *$close = '</p>'*]])[¶](#ihv636)

| Parameters: | * **$open** (*string*) – Error message opening tag * **$close** (*string*) – Error message closing tag |
| --- | --- |
| Returns: | Error messages |
| Return type: | string |

Returns all detected errors formatted as a string.

echo $this->image\_lib->display\_errors();

[Next](http://docs.google.com/input.html)   [Previous](http://docs.google.com/ftp.html)

© Copyright 2014 - 2016, British Columbia Institute of Technology. Last updated on Oct 28, 2016.

Built with [Sphinx](http://sphinx-doc.org/) using a [theme](https://github.com/snide/sphinx_rtd_theme) provided by [Read the Docs](https://readthedocs.org).