

Properly size images



Warning: This audit has moved into the [Improve image delivery](#)

(</docs/performance/insights/image-delivery>) insight as of Lighthouse 13. For more information see [What's new in Lighthouse 13](#) (</blog/lighthouse-13-0>).

The Opportunities section of your Lighthouse report lists all images in your page that aren't appropriately sized, along with the potential savings in [kibibytes \(KiB\)](#).

(<https://en.wikipedia.org/wiki/Kibibyte>). Resize these images to save data and improve page load time:

▲ Properly size images

9.77 s ^

Serve images that are appropriately-sized to save cellular data and improve load time. [Learn more.](#)

☐ Show 3rd-party resources (0)

URL	Size	Potential Savings
 /flower_photo.jpg (correct-dimensions.glitch.me)	1,953 KB	1,918 KB
 /flower_logo.png (correct-dimensions.glitch.me)	71 KB	69 KB

How Lighthouse calculates oversized images

For each image on the page, Lighthouse compares the size of the rendered image against the size of the actual image. The rendered size also accounts for device pixel ratio. If the rendered size is at least 4KiB smaller than the actual size, then the image fails the audit.

Strategies for properly sizing images

Ideally, your page should never serve images that are larger than the version that's rendered on the user's screen. Anything larger than that just results in wasted bytes and slows down

page load time.

The main strategy for serving appropriately sized images is called "responsive images". With responsive images, you generate multiple versions of each image, and then specify which version to use in your HTML or CSS using media queries, viewport dimensions, and so on. Additionally, [ResImageLint](https://ausi.github.io/respimagelint/) (<https://ausi.github.io/respimagelint/>) is a helpful bookmarklet for identifying the optimal `srcset` and `sizes` values for your images. See [Serve responsive images](https://web.dev/serve-responsive-images) (<https://web.dev/serve-responsive-images>) to learn more about these attributes.

[Image CDNs](https://web.dev/articles/image-cdns) (<https://web.dev/articles/image-cdns>) are another main strategy for serving appropriately sized images. You can think of image CDNs like web service APIs for transforming images.

Another strategy is to use vector-based image formats, like SVG. With a finite amount of code, an SVG image can scale to any size. See [Replace complex icons with SVG](https://web.dev/articles/responsive-images#replace_complex_icons_with_svg) (https://web.dev/articles/responsive-images#replace_complex_icons_with_svg) to learn more.

Tools like [gulp-responsive](https://www.npmjs.com/package/gulp-responsive) (<https://www.npmjs.com/package/gulp-responsive>) or [responsive-images-generator](https://www.npmjs.com/package/responsive-images-generator) (<https://www.npmjs.com/package/responsive-images-generator>) can help automate the process of converting an image into multiple formats. There are also image CDNs which let you generate multiple versions, either when you upload an image, or request it from your page.

Stack-specific guidance

AMP

Use the [`amp-img`](https://amp.dev/documentation/components/amp-img/?format=websites) (<https://amp.dev/documentation/components/amp-img/?format=websites>) component's support for `srcset` (<https://web.dev/articles/use-srcset-to-automatically-choose-the-right-image>) to specify which image assets to use based on the screen size. See also [Responsive images with srcset, sizes & heights](https://amp.dev/documentation/guides-and-tutorials/develop/style_and_layout/art_direction/) (https://amp.dev/documentation/guides-and-tutorials/develop/style_and_layout/art_direction/).

Angular

Consider using the [BreakpointObserver utility](https://material.angular.io/cdk/layout/overview) (<https://material.angular.io/cdk/layout/overview>) in the Component Dev Kit (CDK) to manage image breakpoints.

Drupal

Ensure that you are using the native [Responsive Image Styles](https://www.drupal.org/documentation/modules/responsive_image) (https://www.drupal.org/documentation/modules/responsive_image) provided from **Drupal**. Use the Responsive Image Styles when rendering image fields through view modes, views, or images uploaded through the WYSIWYG editor.

Gatsby

Use the [gatsby-image](https://www.gatsbyjs.com/plugins/gatsby-image/) (<https://www.gatsbyjs.com/plugins/gatsby-image/>) plugin to generate multiple smaller images for smartphones and tablets. It can also create SVG image placeholders for efficient lazy loading.

Joomla

Consider using a [responsive images plugin](https://extensions.joomla.org/instant-search/?jed_live%5Bquery%5D=responsive%20images) (https://extensions.joomla.org/instant-search/?jed_live%5Bquery%5D=responsive%20images).

WordPress

Upload images directly through the [media library](https://wordpress.org/support/article/media-library-screen/) (<https://wordpress.org/support/article/media-library-screen/>) to ensure that the required image sizes are available, and then insert them from the media library or use the image widget to ensure the optimal image sizes are used (including those for the responsive breakpoints). Avoid using **Full Size** images unless the dimensions are adequate for their usage. See [Inserting images into posts and pages](https://wordpress.org/support/article/inserting-images-into-posts-and-pages/) (<https://wordpress.org/support/article/inserting-images-into-posts-and-pages/>).

Resources

-
- Source code for **Properly size images** audit
(<https://github.com/GoogleChrome/lighthouse/blob/main/core/audits/byte-efficiency/uses-responsive-images.js>)
 - Serve images with correct dimensions
(<https://web.dev/articles/serve-images-with-correct-dimensions>)

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 4.0 License](https://creativecommons.org/licenses/by/4.0/) (<https://creativecommons.org/licenses/by/4.0/>), and code samples are licensed under the [Apache 2.0 License](https://www.apache.org/licenses/LICENSE-2.0) (<https://www.apache.org/licenses/LICENSE-2.0>). For details, see the [Google Developers Site Policies](https://developers.google.com/site-policies) (<https://developers.google.com/site-policies>). Java is a registered trademark of Oracle and/or its affiliates.

Last updated 2019-05-02 UTC.