51

For `object-fit` to work, the image itself needs a `width` and `height`. In the OP's CSS, the images do not have width and/or height set, thus object-fit cannot work.

The clue: `width` and `height` need NOT be the dimensions of the image itself! Think of it as if it were a `div`: If you want a `div` to fill its container, you will set

```
width:100%; height:100%;
```

...and the browser will know that this div should completely fill its container's space.

In case of an `img`, the browser performs two steps:

1. The browser **creates a bounding box**: By default, the box dimensions will be the exact dimensions of the image itself. But we're free to tell the browser to size the image to 100% of its container's width and 100% of its container's height. Then it will create a box that completely fills the container's space.
2. The browser **fits the image pixels into this box**: By default, the image will be squeezed/stretched so the *image width matches the box width*, and the *image height matches the box height*. But using `object-fit`, you can select **how to match image and box dimensions**. For example, using `object-fit:cover` commands to enlarge/downsize the image to completely fill the box while maintaining its aspect ratio.

Regarding the OP, I would simply set:

```
main > section.posts > article > img {
  width: 100%; /* image box size as % of container, see step 1 */
  height: 100%; /* image box size as % of container, see step 1 */
  object-fit: cover; /* matching of image pixels to image box, see step 2 */
}
```

One final caveat: When using % values for sizing, the container must have a defined width and height for object-fit to work. OP would need to define `height` in `main > section.posts > article`.