# Determining if a web page contains (clickable) images that are wrongfully marked as decorative (and vice versa) using a combination of basic logic and computer vision

Sources:

- Technieken om afbeeldingen een tekstalternatief te geven (AnySurfer)
- Welk tekstalternatief voor afbeeldingen (AnySurfer)
- An alt Decision Tree (W3C)

## 1 - Collect (contextual) information about the images

For a given accessibility tree (that is derived from a web page DOM), we first need to create a list of all elements that have a role of **image**. For each image, we need to collect some information to work with.

(Note that this list is not limited to **<img>** elements.)

### **a** - Visual properties of the image

- Does it contain a (bitmapped) representation of text (OCR)?
    - Yes/no
- Image classification
    - Photo
        - List of keywords
    - Screenshot
    - Illustration

### **b** - Accessible Name

The value of the accessible name is derived from a **visible** (e.g., the visible text on a button) or **invisible** (e.g., the text alternative that describes an icon) property of an element (DOM node) on the page.

This is how it is computed: W3C Accessible Name and Description Computation 1.2.

For example, in **<img alt="foo" aria-label="bar">** the Accessible Name of that element is “bar” because **aria-label** takes precedence over **alt**. There are other rules as well. We don’t need to recreate these computation rules. Use Puppeteer or headless Chrome to query the accessibility tree.

### **c** - Is it marked as decorative or “not important to AT” by the author or not? (true/false)

Look for a null (not missing) **alt=""**, **aria-hidden=”true”** or **role=”presentation”** or retrieve this information from the accessibility tree.

### **d** - Is it interactive (is it clickable or focusable)? (true/false)

Is the image contained in a parent node that is a **link** or a **button** or can it be focused and/or clicked?

Example (clickable image)

```
<a href="…"><img alt="…"></a>
```

(The most reliable way to determine this is by trusting the information that the accessibility tree provides.)

### **e** - Image type

- Bitmap image (JPEG, WEBP, PNG etc.)
- SVG
- Icon font

*(if it is a SVG, it is very unlikely that it is decorative.)*

**f** - Image dimensions (width+height in pixels)

*(if it is smaller than 64x64 pixels or if it has a 1:1 ratio, it is very likely that it is an icon.)*

## 2 - Assumptions (“principles”)

Our assumptions about decorative images:

1. Images that contain text and/or are identified as a screenshot (a) **can never** be decorative.
2. Images that you can interact with (d) **can never** be decorative (buttons, image maps.
3. Images that are of a certain image type (e) or a certain size (e) **are likely** to be **not decorative**.
4. Images that contain no **identifiable content** (using computer vision) are likely to be decorative.
    - However… images that contain **identifiable content** might be decorative as well.
5. (More to add?)

## 3 - Decide

With this information, we can start building a simple decision tree.

Questions to ask:

- Is it marked as decorative by the author (c)?
    - If **yes**, determine how likely it is that that is indeed correct
        - Use assumptions
    - If **no** (= it has an Accessible Name that is not null)
        - Use assumptions