Data analysis

Analysis of rooftiles
Image dataset visualization and preprocessing

Template created by:

Christian Wanschers
Sara Eftekhar Azam
Bastiaan Verheul
Nino van Alphen

March 11, 2024

Table of Contents

1 Visualizations of Dataset

First, it is important to analyze and visualize the properties of the original images. We did this with the following steps...

1.1 BGR channels of original dataset

[1]

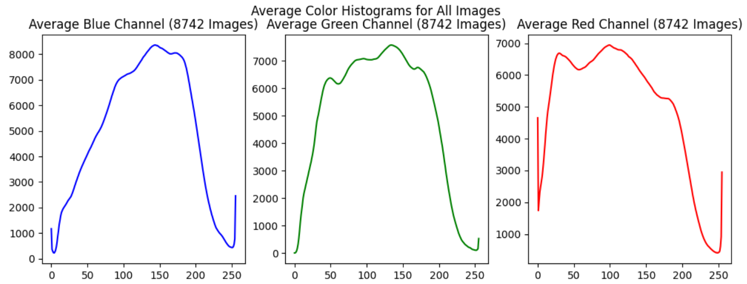

Figure 1: Average colors as colorchannels.

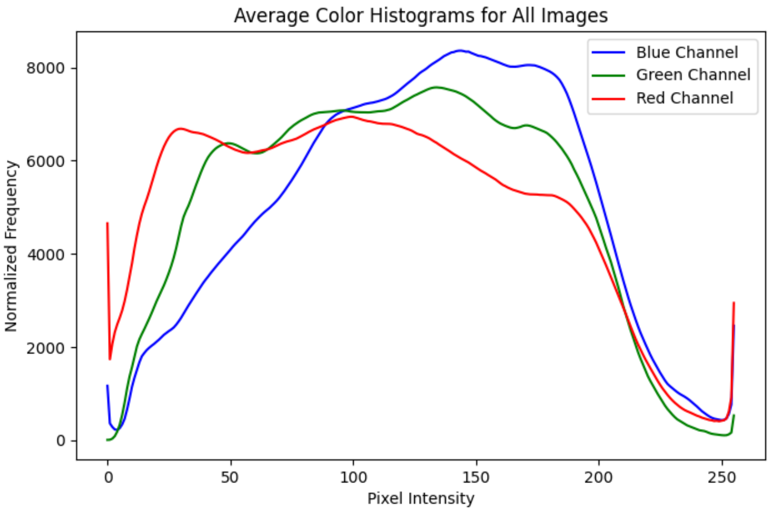

Figure 2: Overlapping colors as colorchannels.

[1] We needed to analyze the color distribution of the colors to understand the average colors used within the dataset. The figure below shows the result of this analysis for the whole dataset and below that examples of 3 random images and their own color channels.

1.2 Visualizing aspect ratios

[2]

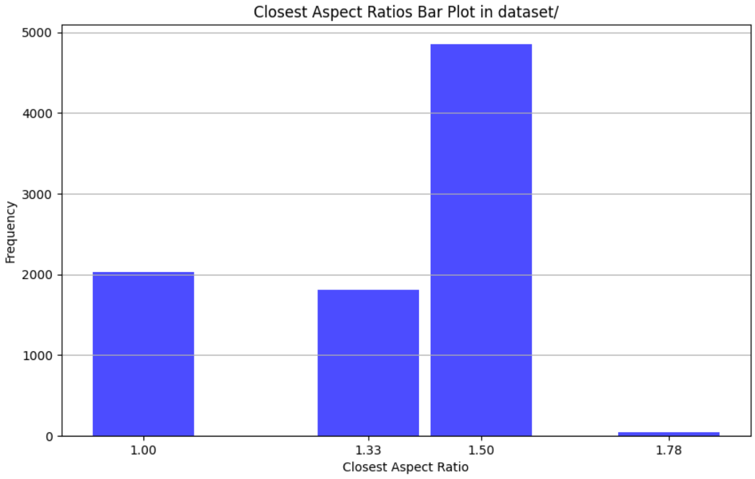

Figure 3: Aspect ratios.

[2] We also need to check the aspect ratios to check whether non-square images were present in our dataset. And as you can see, not all have a square aspect ratio. However, this is required for most and simpler YOLO algorithms

1.3 Visualizing resolutions

Before changing the properties of the images, we also need to check the resolutions to determine if we can safely decrease it to make it easier to work with the model without long loading times which comes with using big files. All images in the dataset we got from the client had roughly had a 4K resolution, which meant each image was around 4- to 6MB in size. This meant the total dataset was around 75GB for 7800 images. The size means that the dataset is difficult to process in code and the model generally does not require images with a resolution that large. We decided to scale the images down when the resolution was of a higher value than 3000 in either width or length. This meant a file size decrease of about 94%. This resulted in the total size of the dataset decreasing to less than 4GB and thus being much easier to process. The figure below shows the result resolution rescaling.

3

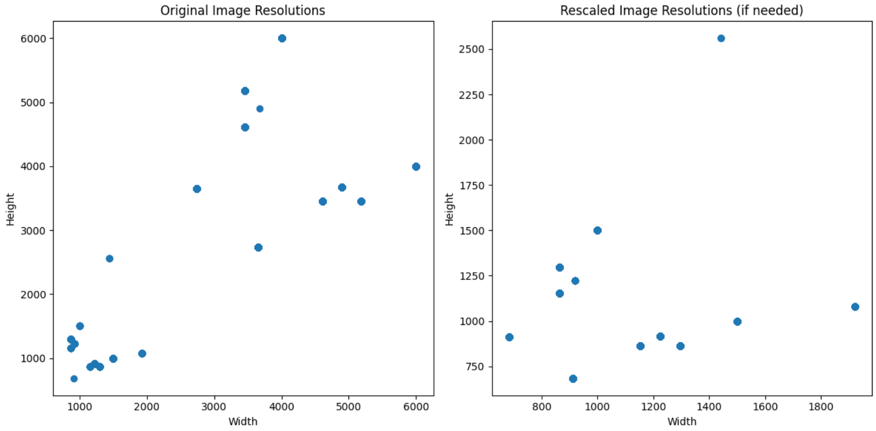

Figure 4: Resolutions of images before & after

[3] Rescaling was done immediatly since the 8742 images within the dataset resulted in a size of over 75GB which isn't easy to work with.

2 Changes to Dataset

We need to change some properties to prepare the dataset before feeding it to a model…

2.1 Updates to resolution

As we mentioned earlier, we decreased the resolution of all images in the whole dataset to make it more manageable. See 4.

2.2 File extension formatting

Most images were of .JPG format and some of .JPEG and .PNG format. To make all these images the same file format we chose to convert them all to the PNG format.

2.3 Applying grayscale

[4]

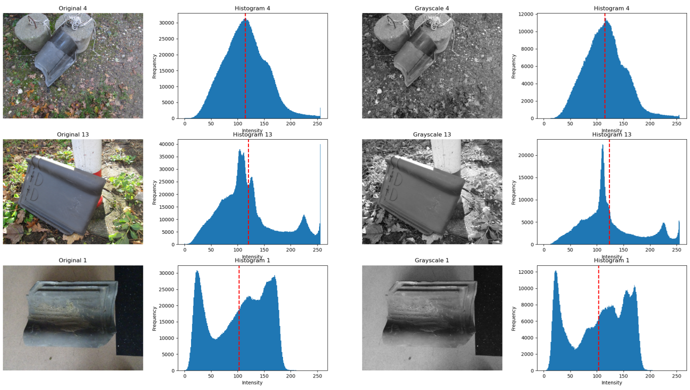

Figure 5: Grayscale of images plot

[4] We need to apply a grayscale effect to remove the BGR-channels currently present in the image. The image contains colours which might interfere with classification and makes it harder to bring out the details. Applying Grayscale makes sure those details are easier to bring out in the next preprocessing techniques.

2.4 Applying normalization

[5]

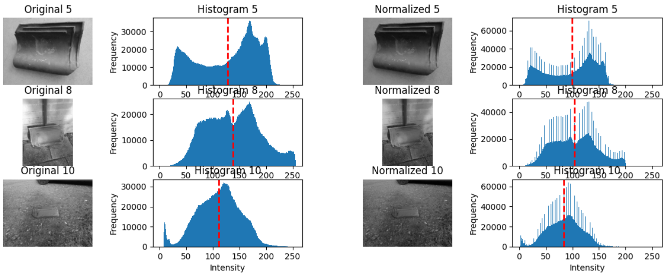

Figure 6: Normalization of images plot

[5] After grayscaling, normalization needs to be applied to normalize contrast. Doing this removes some glare or extreme lighting effects caused by the camera lens.

2.5 Applying CLAHE

[6]

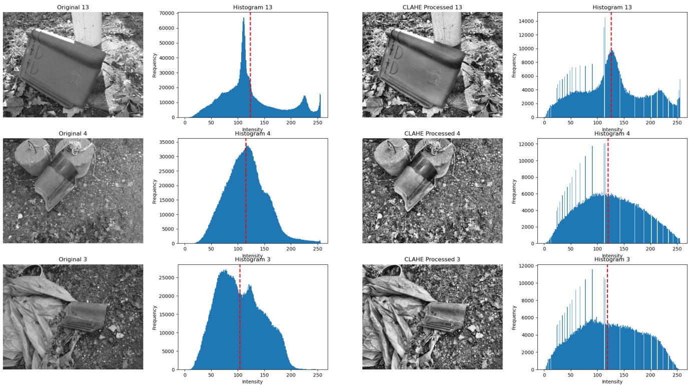

Figure 7: CLAHE of images plot

[6] Lastly, CLAHE is applied to bring out more details. CLAHE does this to dampen some of the higher pixel values and placing them below the average threshold, so that details appear more sharp.

2.6 Updates to aspect ratio

Since most image recognition models require the images in the dataset to be of square format, we came up with a technique to make non-square images square. his is done by placing padding with a random color around the image to fill up the blank space required to make it square. After this a new resolution is immediately applied of 640*640 pixels. This resolution is required by most YOLO-models which could be used as the main algorithm classifying the rooftiles. To the side are a few examples of images converted to a square aspect ratio.

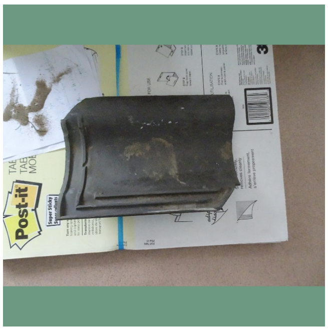

Figure 8: Square1

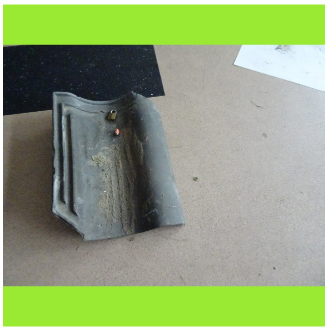

Figure 9: Square2

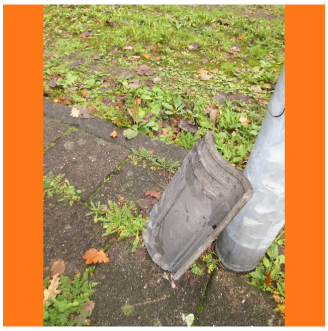

Figure 10: Square3

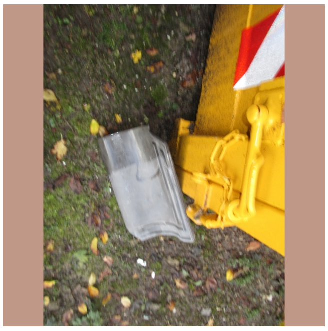

Figure 11: Square4

2.7 Data augmentation

A few augmentation techniques are applied to artificially increase the dataset and training data. It's important to realize that this is just done for training and not testing, since testing requires the images to be as closely to the real world as possible. Below are a few simple techniques which can be applied to any image in the dataset:

2.7.1 Flipping

2.7.2 Rotating

2.7.3 Mirroring

2.7.4 Zoom

2.8 Splitting data

2.8.1 Training

Test

2.8.2 Testing