# 

# 

# 

# Image Processing

## Project results

*Made by:*

**Tim Bakx** 3695301

**Roeland Krijgsman** 3832171

## Subject of the assignment

The target objects for our image processing assignment are LEGO bricks.

### LEGO groups

Since our assignment plan, we’ve had to alter a few of these groups. We’re doing three filter steps, one for each group. We’re grouping them based on their rectangularity.

#### Group A: rectangles

This is your run-of-the-mill LEGO block type. We’re also including the flat *plates* of LEGO, and the higher LEGO bricks. These bricks are represented by the 1x2x2 brick, and the 4x12x⅓ and 2x16x⅓ plates. We’ve also included group C from the original assignment plan (usually rectangular or almost rectangular, not covered all the way with studs).

#### Group B: rectangles with extra

This is the first advanced type of LEGO brick. Mostly used to connect different types of LEGO, these are rectangular bricks with a little extra (e.g. a connector rod). We’ve also had to include group E, as they are found with the same set of parameters. Group E consists of differently-shaped LEGO bricks. These include the housing/roof bricks, and slanted bricks.

#### Group D:

This group consists of distractor objects – either they’re too specific, or they’re things like LEGO minifigures, minifigure accessories and non-LEGO objects.

## Context

### Orientation

LEGO bricks look quite different when upside down. We’re only interested in looking at upright LEGO bricks, so we can use the studs – the little round bumps on top – for detection. Furthermore, we’re excluding bricks lying adjacent to each other.

### Background

We’re using a black background, which is not without noise. There are different lighting backgrounds (some have more of a gradient background), shadows, bits of dust, etc. These will all have to be filtered out when detecting the different defined groups.

### Perspective

We’re using a top-down perspective. As noted previously, we don’t care much about size. We mostly focus on **type** and **proportion**.

## Excluded bricks

We’ve excluded some types of difficult LEGO bricks. These are deemed too difficult to label for this assignment. These bricks all have elements that are so different from the other groups, that we don’t think it’s feasible for this assignment to label these as well as the other groups. We’re adding the darker LEGO block to these, as it’s hard to detect due to light reflecting off the curved side.

## Algorithm

### Grayscale

First, we detect the background and grayscale the whole image. We’ll filter out basic imperfections in this step, using methods like Gaussian blur

### Thresholding

We’ll use Closing/Windowing to select interesting objects/areas

### Watershed/similar methods

We’ll use Watershed or combinations of other methods like Hough transform and others to detect individual edges/borders of objects, and find objects by using these edges

### Edge object removal

Remove all objects that cross the image boundary (as we can’t determine whether they are bigger outside of the image, and don’t want to make assumptions about the rest outside of the image)

### Labelling

We’ll start labelling the objects using our five defined groups

### Filter

Filter the resulting image from the previous steps using the labels. Map these label values to grayscale values, so the groups can be seen in the image

### Original with result

Using the previous step, highlight the selected group in the image by grey-scaling the original image for every pixel except for those that are contained within the target group (keep original colour value here).