

# A simple detector for crowd counting using OpenCV and Python

Martí Gelabert Gómez

University of the Balearic Islands

November 25, 2022

- 1 Background Removal
- 2 Binarization
- 3 Dilation
- 4 Find contours
- 5 Real Matching
- 6 Conclusions

# Procedure

- 1 Import images as black and white.
- 2 Apply Adaptive Histogram equalization to the images.
- 3 Subtract the background to the images using the image with the background.
- 4 Apply an erosion algorithm to obtain small highlighted areas.
- 5 Apply a thresholding algorithm to binarize the image.
- 6 Apply a dilation operation into the binarized images to expand the whites.
- 7 Use a contour algorithm to extract the different regions containing persons.
- 8 Obtain the bounding boxes from these regions.
- 9 Count them and compare the number of detections to the real quantity.

# Background Removal



We need to do something with the mood of the images.



# Background Image



# Gaussian Blur

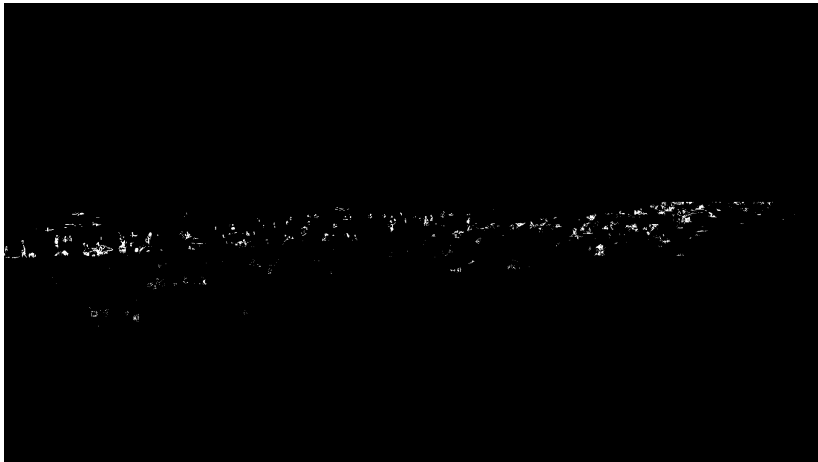


# Substraction





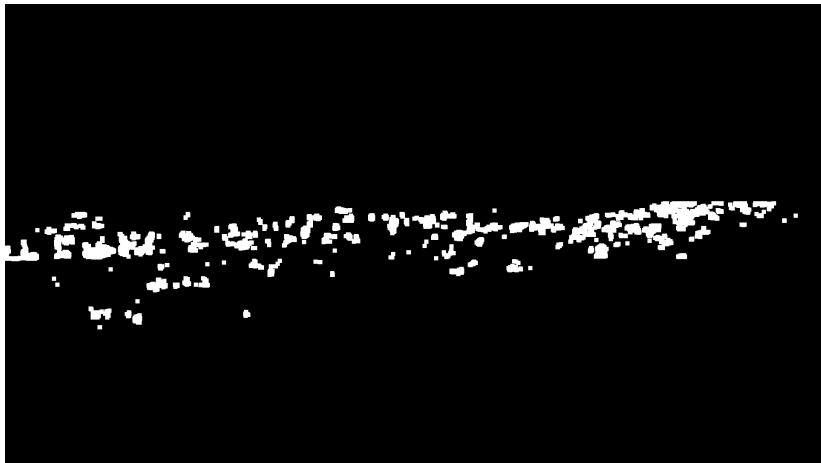
# Binarization



# Masking



# Dilation



# Find contours



# Matching



# Results

**Table:** Performance metrics using the proposed algorithm (withouth using the empty image)

files	preciission	recall	f1 score	gt	detected	match
1660309200.jpg	0.529	0.700	0.603	90	119	63
1660302000.jpg	0.446	0.563	0.498	103	130	58
1660294800.jpg	0.535	0.736	0.620	72	99	53
1660320000.jpg	0.770	0.770	0.770	135	135	104
1660287600.jpg	0.275	0.647	0.386	17	40	11
1660298400.jpg	0.663	0.594	0.627	106	95	63
1660305600.jpg	0.595	0.653	0.623	101	111	66
1660316400.jpg	0.888	0.856	0.872	139	134	119
1660291200.jpg	0.667	0.654	0.660	52	51	34

# Conclusions

It is not good... but at least does something.