11761 - Analysis of Images and Videos
Intelligent Systems
Universitat de les Illes Balears

PROJECT 1: CROWD MONITORING

Yolanda, Amelia
Alsatouf, Abdulrahman

Supervisor:
Ayed, Inés

1st Semester

December 12, 2023

Contents

1 Introduction

In this project we are exploring the topic of object detection within images, in particular for the detection of persons with the non-machine learning methods for image processing. Our main task is to define whether existing image analytics techniques can proficiently estimate the number of individuals in a given image. Therefore, a crowd counting algorithm based on basic image processing techniques such as windowing, thresholding, morphological operations, and edge detection is proposed and developed with the primary objective of providing reliable real-time estimations of the occupancy levels of beaches. This report serves as a thorough documentation of the project, encompassing essential sections including introduction, dataset and annotations, implementation, results and accuracy, and a concluding reflection.

2 Dataset and Annotation

The dataset used in the crowd counting algorithm consists of 10 images retrieved from a security surveillance camera in a beach. The timeframe of the images are varied since there are different lighting and shadows composition in it. There are also various crowds that are spotted on the beach, with at least two people detected in each image. We are using the suggested website to make the annotation. Here we mark the head of a person with a dot, and this dot is saved as a comma separated value containing the coordinate of the dot in the particular image. We did the annotation on all images, and this annotation is later used for the validation purpose.

3 Implementation

Our implementation is divided into the following tasks: image pre-processing, morphological operations, edge detection, contour detection, and person counting. The following image is used for the next parts as an example of parts of the algorithms.

Figure 1: Original image 1660298400.jpg

3.1 Image Pre-Processing

The preprocessing part started by separating the images to avoid capturing the background noise. We defined the cropped area to be the upper part of the image where there is a small chance of detecting people; this includes the area of water, mountains and clouds. We cropped the image by defining the starting and ending indices along each dimension (rows and columns) of the images, and then saved the cropped region and remaining region into separate values.

Figure 2: The image 1660298400.jpg after applying grayscale

After that, we convert the image into grayscale value considering the image processing algorithm that works better with one color channel. Then we applied a windowing function to normalize the pixels and increase contrasts of the image. This task is done by clipping the pixel values to be within the specified window. The figure below shows the result after we applied the windowing function.

Figure 3: The image 1660298400.jpg after applying windowing

For the next part, we are applying Otsu's thresholding and then inverting it to negative. Otsu's thresholding is a method used to automatically determine the optimal threshold value for separating the pixels of an image into separate classes. The image is then thresholded based on the calculated optimal threshold, resulting in a binary image where pixels with intensity values above the threshold are set to one color, and pixels below the threshold are set to another color.

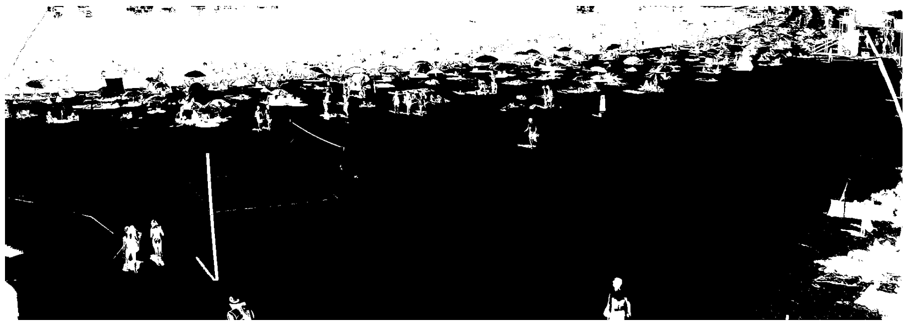

Figure 4: The image 1660298400.jpg after applying the Otsu's thresholding

3.2 Morphological Operations

The crowd counting algorithm incorporates morphological operations to enhance and refine the segmentation of persons within images. Morphological operations are fundamental image processing techniques that involve the manipulation of image structures through the use of structuring elements.

At the beginning, the algorithm employs morphological opening to address noise and small-scale variations in the image. The opening operation involves the application of a structured element to erode and then dilate the image. This helps in smoothing contours and separating objects that may be close to each other.

Additionally, closing is utilized to connect small white regions or gaps in the image, contributing to the consolidation of relevant structures. The closing operation consists of dilating the image followed by erosion. It is particularly effective in closing gaps in contours and making the objects more compact.

Both operations are deployed in Python by calling the function `cv2.morphologyEx()`. As an illustrative example, the output image after applying morphological opening and closing is shown in Figure 5.

Figure 5: The image 1660298400.jpg after applying morphological operations

3.3 Edge Detection

After deploying morphological operations, the proposed algorithm makes use of edge detection techniques. Particularly, Sobel edge detection is employed to enhance the visibility of edges within the preprocessed image [1]. The application of Sobel edge detection allows to highlight significant edges within the image, which aids in the identification of contours related to persons. Sobel operators are applied along both the x and y directions to compute the gradient magnitudes. The resulting edge map is then thresholded to emphasize edges of importance for crowd counting. The Sobel edge detection is applied in Python using the function `cv2.Sobel()`. The detected edges by Sobel on the image 1660298400.jpg are shown illustratively in Figure 6.

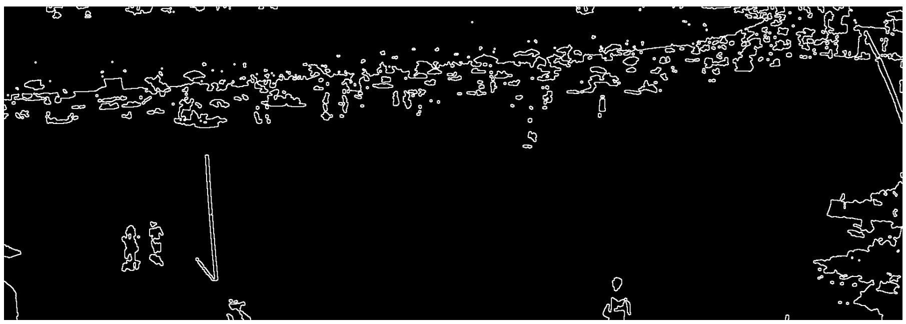

Figure 6: The image 1660298400.jpg after applying Sobel edge detection

3.4 Contour Detection

Finally, contours within the processed images are identified based on the edges obtained through the Sobel edge detection. These contours represent potential regions of interest that may correspond to individuals within the image. A mask is created to outline the areas associated with these contours. The contour detection is carried out in Python by deploying the fucntion `cv2.findContours()`.

Following this, a filtering process is applied to the contours. Contours are scrutinized based on certain criteria, specifically, their area and aspect ratio. Contours with areas falling within the range of 200 to 800 pixels and aspect ratios between 0.4 and 4 are retained. This selection mechanism ensures that contours resembling individual persons are considered, while others are discarded. The resultant filtered contours are the most crucial part of the crowd counting approach deployed by the algorithm, as they are used primarily for identifying individuals within the processed images. Figure 7 shows the filtered contours detected on the image 1660298400.jpg.

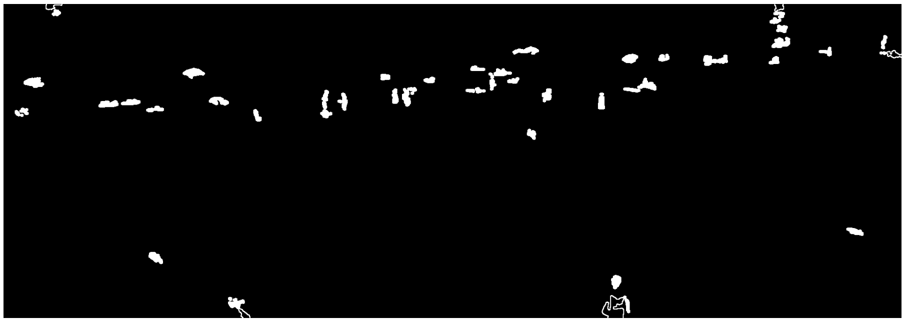

Figure 7: The image 1660298400.jpg after applying contour detection

3.5 Person Counting

In order to obtain the final output of the algorithm, the remaining part of the image is restored and the potential individuals represented by the detected contours, on the region of interest, are marked and counted. Bounding boxes are drawn around these contours, including their centroids. Optionally, extended bounding boxes can be created for comprehensive coverage. The algorithm then compares centroids of the drawn bounding boxes with annotated person coordinates using the Euclidean distance for detection. Detected persons are counted and visualized by overlaying bounding box centroids on the restored original image. The process provides estimates based on spatial relationships between detected contours and annotated ground truth data.

4 Evaluation

The evaluation of the algorithm's performance was conducted at both image and person levels. For each image, the number of (manually) annotated persons ranged from 2 to 79. The algorithm's estimates, including both the number of detected persons and true positives are compared to the number of annotated persons, as can be seen in Table 1. The Mean Squared Error (MSE) across all images was calculated as 3097.6.

At the person level, the detected number of persons, representing true positives, varied across images. The average precision, indicating the proportion of correctly detected persons among all estimates, was computed at 0.6606. Meanwhile, the average recall, reflecting the fraction of annotated persons successfully identified by the algorithm, was determined to be 0.3637. However, the precision and recall per image show an interesting behaviour when compared to the number of annotated persons, as they tend to be higher in images with larger numbers of persons (see Table 1). This might be due to the fact that the brightness of images that exhibit larger numbers of persons is likely to be higher which significantly facilitates the detection process for the algorithm (see Figure 8).

Additionally, the correlation coefficient between annotated and detected persons across all images was found to be 0.8611. This metric signifies a strong positive correlation, suggesting the algorithm's effectiveness in capturing the overall trend in person counts.

Image	Annotated	Detected	True positive	Precision	Recall
1660284000.jpg	2	4	0	0.0000	0.0000
1660287600.jpg	15	18	7	0.3889	0.2692
1660291200.jpg	34	17	8	0.4706	0.1860
1660309200.jpg	52	29	26	0.8966	0.4727
1660302000.jpg	56	34	28	0.8235	0.4516
1660294800.jpg	56	26	18	0.6923	0.2812
1660298400.jpg	58	39	37	0.9487	0.6167
1660320000.jpg	64	59	43	0.7288	0.5375
1660316400.jpg	64	45	35	0.7778	0.4730
1660305600.jpg	79	33	29	0.8788	0.3494

Table 1: Evaluation results of the proposed crowd counting algorithm

Figure 8: The image 1660298400.jpg after applying the proposed crowd detection algorithm

5 Conclusion

This project allows us to learn more on the topic of crowd counting with some image processing techniques. We did a lot of trial and error to reach the current result, and get to observe that using centroid of the bounding boxes allows us to obtain more accurate results (compared to a general bounding boxes). Based on the evaluation performed, we have received a considerably good result with the crowd counting algorithms. However, we also take into consideration the suggestion that were given at class by José María, that it is better to use region instead of contour prior to running the crowd counting algorithm. It is a valuable insight and we will consider this for our next projects. Also for the future improvement, we might want to consider more complex algorithms or techniques such as machine learning for the implementation of object detection.

The source code for this project can be found here.

References

[1] O. Vincent and O. Folorunso, “A descriptive algorithm for sobel image edge detection,” 01 2009.