Detection and Analysis of paintings and people in an art gallery

Manuel Iaderosa, Davide Previati, Riccardo Piccolo

Abstract

This paper presents a pipeline for video analysis carried out in the Galleria Estense in Modena. The pipeline allows, thanks to the use of a neural network, to recognize paintings and people in the recorded rooms. Then using techniques of computer vision tries to correct the perspective distortion of every recognized painting, re-projecting it in front of respect to the image plane. The rectified version of the painting is then used by a matching algorithm to find its similarity with a database of paintings trying to find the greatest similarity for the painting corresponding to the one analyzed. The latter information comes finally exploited to indicate, for the individuals identified, in which gallery room are located. The pipeline is meant to reach the highest possible performance in terms of accuracy, both in the detection of paintings and people, both in painting rectification operations and retrieval to completely carry out all the tasks at each frame analyzed.

1. Introduction

Among the various fields covered by computer vision, object detection is definitely one of the most important. It is the starting point for applications of any kind, providing the basis for scene analysis and understanding. Pure computer vision algorithms perform filtering, edge detection, template matching etc. can be applied in order to isolate the object of interest, but they can be very prone to noise problems, changes in image lighting etc. Nowadays deep learning methods who use convolutional neural networks outperform these computer vision techniques in terms of accuracy, having become now the state-of-the-art in the object detection task. In the case of this project a pipeline, capable of analyzing the recordings taken inside an art gallery, is created, recognizing paintings and people among all the elements present in the scene using a neural network. We then try to improve the results obtained by object detection, giving an identity to the elements identified through a tracking procedure. In this way, it is possible to keep track of movements over time of every single object. The paintings detected are then rectified for correct their perspective distortion and matched with a database of paintings in order to find to most similar one. This information is then exploited on people detected to predict in which room of the gallery they are.

2. Related Works

This section discusses related work for object detection, tracking and object recognition techniques

2.1. YOLOv3 for Object Detection

Object detection is a domain that has benefited immensely from the recent developments in deep learning. Recent years have seen people develop many algorithms for object detection, some of which includes YOLO, SSD, Mask RCNN and RetinaNet. One of the fastest and most used object detection algorithms is YOLOv3. It uses features learned from a deep convolutional neural network to detect an object. It is composed of a series of convolutional layers, with skip connections and upsampling layers. It does not use pooling to prevent the loss of low-level features. The downsample of the images is done through a stride factor. The neural network output is a feature map that is passed to a layer 1x1 convolutional to obtain a prediction map of the same size [1]. This prediction map indicates that each cell (neuron) can predict a fixed number of bounding boxes, in this case 3. There are in all (B x (5 + C)) entries in the feature map. B is the number of bounding boxes that each cell can provide (3), and each bounding box has 5 + C attributes, i.e. the two coordinates of the center, width, length, objectness score and C class confidences for each bounding box. The objectness Score indicates the probability that an object is contained within a bounding box. Should be close to 1 for the responsible cell and its neighbors, and close to zero for the edges of ground truth. It is passed to a sigmoid for make the objectness score a probability. By class confidences instead we mean the probability that the detection belongs to a particular class, this one also in sigmoid. In YOLO training, it is chosen responsible for the detection of a single object the cell that falls in the center of the ground truth of the labeled object. This cell can then provide 3 bounding boxes. Width and height of the bounding box are not directly calculated (unstable gradients in training), but log-spaces trans-

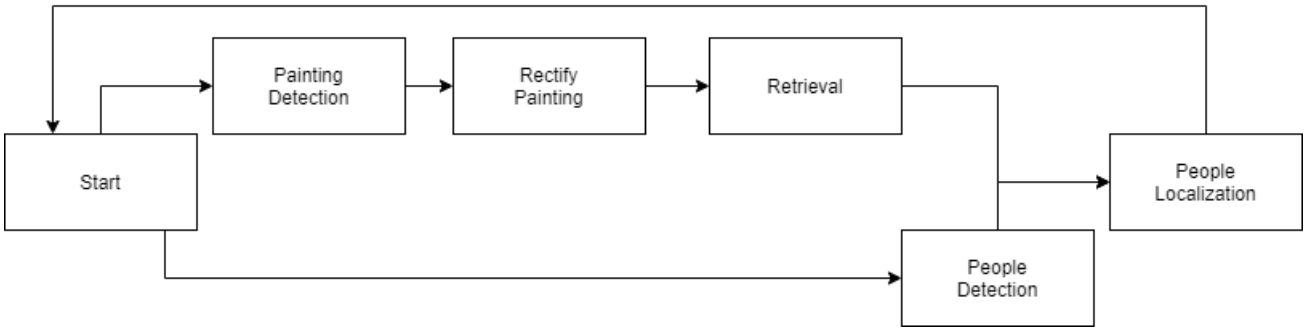

Figure 1. Pipeline of the project

form and offset called anchors are calculated and used to get the prediction. It is then selected ,for the cell responsible, the bounding box whose anchor has the highest IoU with the ground truth box. In addition, YOLO predicts on different scales, therefore feature maps with different sizes are created with different anchors for each scale. It results in better detection of small objects [2]. Finally, to reduce the size of the output detection, a thresholding based on the objectness score and Non Maxima Suppression is used to filter the detections on the same image.

2.2. Tracker SORT

The object detection phase can be further improved by adding a tracking technique to identify the bounding boxes and keep track of them between the various frames. It is an example the SORT algorithm (Simple Online Realtime Tracking) composed of the Hungarian Algorithm to indicate if an object in a frame is the same as the previous frame, and by a Kalman Filter to predict the future position based on the current one, to have better associations [3]. More precisely, the Hungarian Algorithm (Kuhn-Munkres algorithm) associates the detected objects in different frames by measuring scores based on IOU and shape (bounding boxes of similar size). This creates a matrix that indicates the matching between Detection and Tracking, from which it can understand which element in detection corresponds to the element in tracking, and assign to it a unique id. The Kalman Filter then mainly performs two tasks on each bounding box: predict and update. It tries to estimate two values: state mean, or the state vector that indicates the coordinates of bounding boxes and respective rates of change (initialized to 0), and a covariance matrix which indicates uncertainty in the estimate. So predict uses the covariance matrix to predict the bounding boxes of frame t based on the bounding boxes of frame t-1. The update phase is instead a correction step that measures the error between forecast and the original measurement and improves the phase of predict achieving even better results than YOLO in obtaining the bounding boxes values.

2.3. Feature matching

Features matching is the task of establishing correspondences between two images. A common approach to feature matching consists of detecting a set of interest points each associated with image descriptors from image data. Once the features and their descriptors have been extracted from two images, the next step is to establish some preliminary feature matches between these images. Generally, the performance of matching methods based on interest points depends on both the properties of the underlying interest points and the choice of associated image descriptors. We have decided to choose ORB (Oriented FAST and Rotated BRIEF) that is basically a fusion of FAST keypoint detector and BRIEF descriptor because these techniques made it an algorithm with good performance mostly in computation cost, in matching performance and now it is an efficient alternative to SIFT and SURF [4]. After finding the keypoints we have realized the feature matching with Brute-Force Matcher that takes the descriptor of one feature in first set and is matched with all other features in second set using some distance calculation and the closest one is returned.

3. System Overview

Figure 1 shows the functioning of the pipeline, where the main components are described in this section. It can be noted five tasks performed sequentially.

3.1. Painting Detection

Each frame analyzed is sampled in a size of 416x416 to be correctly computed by the neural network. A YOLO neural network model is then used for detection, trained on the class of paintings only. Evaluation is then done and the bounding boxes of each painting identified in the image are calculated. The calculation of bounding boxes are then further refined using the SORT tracker for two reasons: avoiding that in two consecutive frames bounding boxes of too different sizes are estimated for the same object if the scene does not change and to keep track of the identities of rec-

Figure 2. Example of frame when paintings are detected

ognized objects to allow better representation of data. The output of this first task are therefore, for each frame, the coordinates of the bounding boxes of each recognized and labeled painting as shown in Figure 2.

3.2. Painting Rectification

After the painting detection, the portion of the image inside the bounding boxes containing the identified painting is selected, used as input of the following task. First of all the image is converted into grayscale, then 2 dilate cycles are carried out to join any discontinuities in the contours of the painting. Blurring and thresholding techniques are applied to best clean the images from noise and any shadows, and contours of the image are calculated. Starting from the identified contours, the contour with the largest area is selected and, among these, the points that form the smallest convex figure that contains them are chosen. The points that form the perimeter of the main figure (in this case the painting) are then identified and among these the 4 corners to be used to project the picture frontally on the image plane are selected. A more generic formulation is used to manage non-squared paintings as well, which selects as 4 corners the external points calculated in this way (example to select the top-left corner):

$$tl1 = \underset{x,y}{\operatorname{argmin}}(2x + y), \forall (x, y) \in Perimeter \,(1)$$

$$tl2 = \underset{x,y}{\operatorname{argmin}}(x + 2y), \forall (x, y) \in Perimeter \,(2)$$

$$tl = (min(tl1_x, tl2_x), min(tl1_y, tl2_y)) \,(3)$$

Formula (1) selects the leftmost point, also giving weight to the height. Formula (2) selects the highest point, also giving weight to the width (to the left). Thus the 2 points identified are used to select the corner with formula (3) as a combination of the coordinates of the two points. In the case of square paintings the 2 points identified often correspond and correctly identify the corner to be used for the projection. Once the 4 corners have been identified, the 4 projection points (the 4 corners of the output image) are chosen

Figure 3. Example of sequence of operations, from left to right, for painting rectification

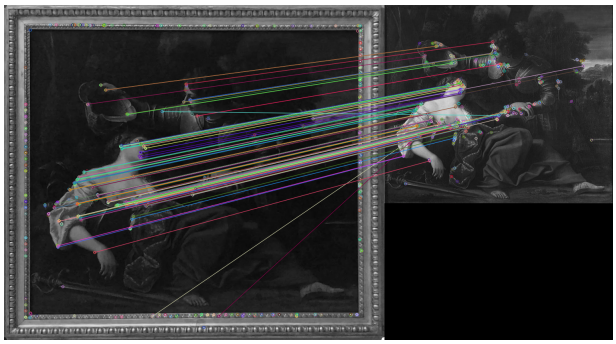

Figure 4. Example of keypoints matching between painting input and painting in database

and, after obtaining the transformation matrix between the correspondences, the projection is carried out. Finally, to recover the information on the depth of the picture, lost due to the perspective distortion, an approximation is used based on the pairs of parallel sides constructed starting from the 4 corners identified. More precisely, the ratio on each pair of sides is calculated and the value obtained is multiplied to obtain depth in width, and in the other case, in height. Figure 3 shows all the operations described.

3.3. Painting Retrieval

For painting retrieval, we have loaded the paintings of the database provided and as input we have taken the painting that was rectified in the previous task. Then an ORB object was created, which we will need to find the keypoints and descriptors. Then, for each painting in this database, we have found the keypoints, that are the positions where the feature have been detected, and their descriptor, that is an array containing numbers to describe that features and we have compared them one by one with the input painting

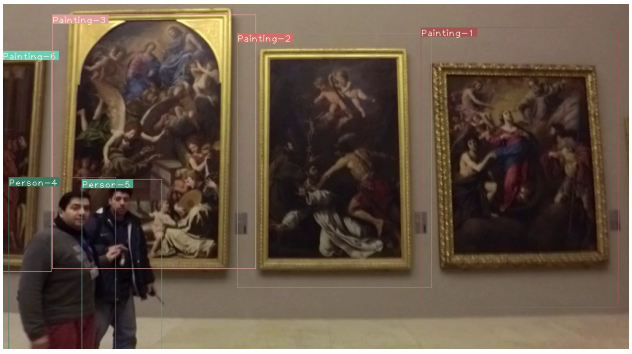

Figure 5. Example of frame when people are detected

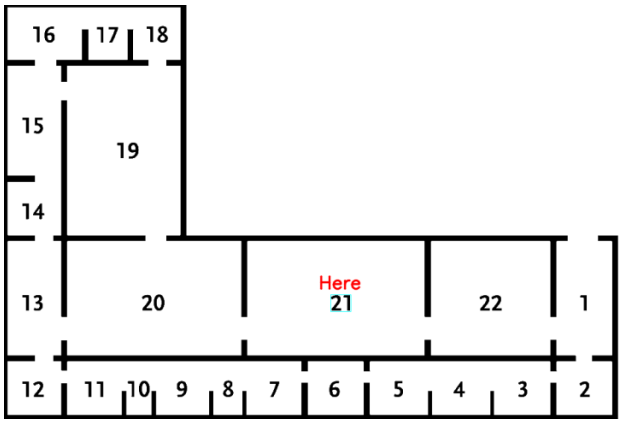

Figure 6. Example of person that is located in a specific room based on painting retrieval

through Brute-Force Matcher, which we have created specifying NORM HAMMING as distance measurement, in particular we have used as matching method knnmatch to get k best matches and we have chosen k=2 so that can apply ratio test (see Figure 4). Continuing we have compared the distances between the 2 matches looking if their ratio had been less than a constant chosen after a discrete analysis regarding the amount of positive matches found and the possible values of the constant and we have calculated the percentage of similarity by making the relationship between the matches positives of the two paintings and the minimum of keypoints found between the 2 paintings. In conclusion, we have made a ranking list in descending order of the paintings with more similarities than the input painting.

3.4. People Detection

Each frame analyzed is sampled in a size of 416x416 to be correctly computed by the neural network. Here another more complex neural network is used for detection of people, choosing the pre-trained model provided by the official YOLOv3 documentation used for multi-class classification out of 80 classes, including people. All the results are then filtered except those belonging to the Person class. Here too, as in section 3.1, tracking is used for refine the calculation of bounding boxes and keep track of people's identities over time. To avoid that YOLO could recognize the figure of a person inside a picture we made checks on the bounding boxes of both the painting and people; more precisely the bounding boxes of the person should not be inside the painting bounding boxes. In case it occurs this hypothesis the person detected would be ignored. The output of this task are therefore for each frame, the coordinates of the bounding boxed of each person recognized and labeled as shown in Figure 5.

3.5. People Localization

Localization of a person is carried out when the painting retrieval and detection of a person are present in the same frame. People localization identifies a person's position by indicating a specific room within the museum based on the room in which the painting retrieved is assigned. Once the room in which the located person is identified, his position is displayed in the image that represents the map of the museum. To do this, we started from the image of the map and found the contours of the numbers and then saved the bounding boxes that enclose them in a file called with the number to which the bounding boxes belong. Once the function that identifies the room number on the map is called, the corresponding file will be searched and the bounding box will be created around the number of the room where the person is present as shown in Figure 6.

4. Experiments and results

This section shows a series of analysis carried out on the tasks taken individually, in order to evaluate their actual accuracy on the individual frames. Only videos that were in valid condition for an analysis were taken into consideration, for example there have been discarded those in which there were only paintings not taken entirely or only statues.

4.1. Painting Detection

For the painting detection, a quantitative analysis was carried out, taking into consideration a series of frames starting from some sample videos. Specifically, the number of true positives (correctly identified pictures), false positives (identifications of incorrect pictures) and false negatives (ignored pictures) were counted for each frame. Since it is only the detection of frames, there is no way to provide a measure for the true negatives, so this value is assumed to be 0. To have a more detailed and precise analysis, avoiding to analyze many frames, also very similar to each other, the measures are made on videos containing at least 2/3 paintings, analyzing not consecutive frames. They are not then considered as false negatives or true positive far

PAINTING DETECTION			
TP	TN	FP	FN
92	0	5	3
ACCURACY	PRECISION	RECALL	F1 SCORE
0,92	0,94	0,96	0,94

Figure 7. painting detection results

PAINTING RETRIEVAL			
TP	TN	FP	FN
85	0	15	0
ACCURACY	PRECISION	RECALL	F1 SCORE
0,85	0,85	/	/

Figure 8. painting retrieval results

PEOPLE DETECTION			
TP	TN	FP	FN
88	0	2	10
ACCURACY	PRECISION	RECALL	F1 SCORE
0,88	0,97	0,89	0,92

Figure 9. people detection results

paintings that are found in other rooms. Therefore, these 4 metrics were used to calculate accuracy, precision, recall and f1 score on a total of 100 frames. Results in the table at Figure 7.

4.2. Painting Rectification

For painting rectification only a qualitative analysis is made, as it is difficult to establish evaluation metrics that measure the accuracy of the task. Obviously the rectification operations carried out on the painting detected not entirely visible are not taken into consideration, although several times it still manages to rectify them. For squared painting the task generally performs well, correctly identifying the four corners corresponding to the corners of the frame and then projecting them frontally into the image plane. The main problems occur in case of strong shadows that cannot be completely removed by the filtering techniques, thus being treated as corners and included in the projection. The formula ... served to treat non-squared painting in a generic way. Unfortunately for the circular ones it is not possible to select exactly the 4 external points to project them, obtaining a rectified version that is not completely visible. Other problems can occur for prospectively united paintings, so it becomes almost impossible to recognize the boundaries to obtain the corners.

4.3. Painting Retrieval

For the painting retrieval, a quantitative analysis was carried out by choosing a subset of videos in which the detection found paintings which were present in the database. To have a more accurate analysis, it was decided to analyze frames of the same non-consecutive video but with a temporal distance of about 70 frames and in total an amount of 100 frames were considered in the statistics. In particular, the number of true positives and false positives were counted for each frame. In fact, the true negatives and false negatives have not been counted because having decided to take into consideration only paintings present in the database, in this task either the painting found most similar to the one in input is the right one (true positives) either it is the wrong one (false positives). Therefore, these 2 metrics were used to calculate accuracy, precision, setting true negatives and false negatives to 0. Results in the table at Figure 8.

4.4. People Detection

For people detection, the same quantitative analysis of painting detection was carried out. The number of true positives (correctly identified people), false positives (identifiers of incorrect people) and false negatives (ignored people) were therefore counted on a total of 100 frames. Here too, from the 4 metrics, accuracy, precision, recall and f1 score were calculated. It can be noticed that unlike the paintings, it is easy to run into more false positives, corresponding as detection of patterns of people inside pictures. The pipeline filters those whose bounding boxes fall within those of the paintings, however if the network were to calculate bounding boxes for people who slightly overdo those of the paintings there is no further control and inevitably a false positive occurs. False negatives also occur more frequently, being the patterns that identify people much more difficult to spot than those in the paintings. Results in the table at Figure 9.

4.5. People Localization

For people localization, since the output of the task directly depends on the painting retrieval, the results (as long as you identify a person in the frame) are the same as shown in the table at Figure 8

5. Discussion

At first glance, the use of two neural networks may seem inefficient for the detection of people and paintings, but as specified at the beginning, the pipeline is designed to achieve the highest possible performance in terms of accuracy, even at the expense of computational performance. In this case the training of a neural network for the simultaneous detection of paintings and people could end in two ways: a very long training with a lot of data was essential

to get good performances on people,but it compromised the results on the pictures, where the neural network tried to minimize the size of the bounding boxes, even at the cost of ignoring the picture. On the other hand, a less profound training to favor the detection of paintings resulted in poor accuracy in the detection of people, which certainly needs many more heterogeneous data. Finally, a compromise between the two cases had poor results on both detections, so in the end we opted to dedicate one specific neural network for each of the two detections, obtaining excellent results on both detections. As already indicated in section 4.4, it can happen that people depicted inside the paintings can be detected as false positives. All the detections of people whose bounding boxes fall within the receptive field of those of a painting are then filtered.

Regarding the painting retrieval there are several paintings not present in the database for matching, which involves the task of assigning them with a wrong painting. As a result, the people localization task, which depends directly on this output to find the room where a person is, fails also. It can happen in this case that even in consecutive frames the matcher assigns different pictures, thus changing the person's room of belonging each time.

6. Conclusions

We presented a pipeline for detection and analysis of paintings and people on videos taken inside an art gallery. The use of a neural network for object detection combined with a tracking algorithm, applied for both paintings and people, allows to achieve high results in terms of accuracy and other evaluation metrics. Doing so gives an excellent starting point for subsequent object analysis tasks. Future developments could be to exploit the tracking of objects to avoid repeating the rectification, retrieval and localization tasks on the same entities, thus ensuring, in addition to greater solidity, performance in terms of pipeline execution speed.

References

[1] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi. *You Only Look Once:Unified, Real-Time Object Detection*. University of Washington, Allen Institute for AI, Facebook AI Research, 2016

[2] Joseph Redmon, Ali Farhadi. *YOLOv3: An Incremental Improvement*. University of Washington, 2018

[3] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, Ben Upcroft *Simple Online and Realtime Tracking*. Queensland University of Technology, University of Sydney, 2017

[4] Ethan Rublee, Vincent Rabaud, Kurt Konolige, Gary R. Bradski. *ORB: an efficient alternative to SIFT or SURF*. Willow Garage, Menlo Park, California, ICCV 2011