Panorama Stitching With Foreground Subtraction

Authors : PEETERS Thomas, BERNARD Aurélien, BERTRAND Alexis, FERY Loïs, PIETTE Elias

Abstract

In modern smartphones, panorama stitching has become a commodity in camera applications. However, most of these applications lack the ability to remove foreground objects in the panorama. This creates panoramas where a same entity can obstruct several sections of the background in one same panorama image.

In this project, we use an NVIDIA Jetson TX2 to capture panning image sequences where people and sports balls are moving through the scene. We then leverage classical background subtraction techniques along with an object detection neural network to detect foreground objects and stitch a panorama image with no foreground objects present. In addition, we estimate the depth of the balls relative to the Jetson module.

We assess our movement detection algorithm using confusion matrices. Which, depending on the algorithm, gives us different recall and precision values(detailed in results section). The object detection algorithm is also evaluated using a mixture of intersection over union (IOU) and confusion matrices. We obtain a 100% recall rate and 99% precision for person detection. 85% and 98% respectively for ball detection. These statistics were computed over 124 bounding boxes.

A comparison of the obtained panorama without foreground subtraction, a reference ideal panorama, and the result obtained using our algorithm is also provided to qualitatively evaluate the effectiveness of our technique.

No foreground subtraction panorama stitching.

Our proposed algorithm for foreground subtraction panorama stitching.

Problem statement

The aim of the project is, thanks to an NVIDIA Jetson TX2, to detect and segment motion in a video stream while the camera is in motion through a left to right panning, and to produce a panoramic view of the image background.

The main goal of the project is to remove all foreground object of the video. A way to do so is through background subtraction. By classifying pixels as foreground objects or background, we can easily remove the moving objects. However, as the camera is panning, an estimation of the motion of the camera has to be done to suppress the movement as every pixel is moving. As a result, a segmentation map of the moving objects would be obtained.

Based on the moving object detection, a panoramic view of only the background has to be created as well as a visualisation of where the camera is situated in this view.

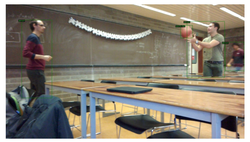
Furthermore, the detection of the two types of moving objects we encounter has to be performed. A bounding box around the balls and people in frame has to be displayed for each image of the video sequence. Also, the 3D position of the balls also has to be tracked. The x and y positions are relative to the frame but the depth has to be an approximation of the real depth of the ball with respect to the camera, in meters.

Finally, we were tasked with developing tests to assess and analyse the performance of our solutions to the previous tasks.

Methodology

1) Image capture:

A sequence of 1500 frames was captured using a Jetson TX2. In these sequences, only sports balls and humans are allowed to move. No moving object must be present in the first frame.



Object detection bounding boxes.

2) Object detection, first foreground approximation :

For each frame and in an online manner, bounding boxes for each person and ball are found. This is done using a fine-tuned pretrained object detection model. To achieve real-time performances, images are downscaled and only one in every 6 frames is processed. These bounding boxes are used as a first approximation of the foreground mask of each frame.

3) Rotation Estimation :

We estimate the rotation of the camera from one image to the next using 2000 SIFT features per image. We use a Brute Force matcher to find their feature matching. To improve matching, the foreground mask is used to exclude features in moving objects. Finally, the essential matrix is derived from the feature matchings, and the camera pose is recovered from the essential matrix.

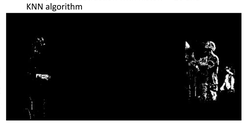
Object detection based foreground mask

4) Movement detection for background subtraction :

Motion detection was done using the KNN/MOG2 background subtraction algorithm. However, to accommodate the fact that the camera was moving, thus the background too, the images we fed to KNN/MOG2 were the current panorama, based on what is to be taken as the background according to KNN/MOG2 at each step.
The masks generated through this process were then filtered thanks to the bounding boxes of moving objects generated earlier, excluding every pixel on the mask that is not a part of a bounding box.
This compensated a bit the errors in the rotation calculation that directly impacted the panorama generation and thus the background subtraction which would consider errors in the panorama evolution as moving pixels.

SIFT features

Background subtraction with history = 20 and KNN algorithm



Background subtraction with history = 20 and MOG2 algorithm

5) Image Warping and Stitching :

To stitch the images, every frame followed the same process:

- First, the homography matrix was computed using the previously computed rotation matrix.
- Then, the images where warped into their cylindrical projection and translated to the corresponding position using the homography matrix. The previously computed foreground mass of the corresponding image was warped in the same manner.
- Finally, using the foreground mask, only the background of each warped image was blended into the panorama image.

Cylindrical warping

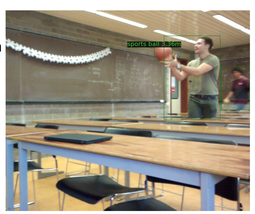
6) Depth estimation :

To estimate the depth with respect to the Jetson module of the sports balls in the sequences, the following equation was used:

$$\text{Depth} = C \times \text{Focal Length} \times \frac{\text{Size of object[m]}}{\text{Size in image[pixels]}}$$

All parameters except C are known or are measured. The unknown parameter C is found by taking pictures of a given ball at different known depths and averaging the resulting C. The found coefficient in our case is C = 1.02 (averaged over 6 images)

The precision of the depth estimation is then tightly related with the precision of the bounding box given by the neural network. Also, as the given balls have different sizes, recognizing the ball in the bounding box was necessary to infer its size. To do so, the average colour in the bounding box is compared it to the average colour of the given balls. The smallest euclidean distance was predicted as the ball type in the box.



Ball depth estimation

Results

1) Moving object detection module

To access the performance of the moving object detection module, we choose to define the mask creation as a binary classification problem where each pixel of an image is classified as “black” or “white”. We choose to define the white pixels as the positive class. We can then compute a global confusion matrix by comparing each pixels of each predicted mask to they corresponding ground truth and summing the following four quantities in a confusion matrix:

TN: Black pixel that are predicted as black pixel.
FN: White pixel that are predicted as black pixel.
FP: Black pixel that are predicted as white pixel.
TP: White pixel that are predicted as white pixel.

Pred/True	F	T
F	3 6323e+07	37 188
T	4728	9185

Mean confusion matrix for the MOG2 algorithm computed on the segments our group annotated. We computed them on whole panoramas, thus the high true negative value.

Pred/True	F	T
F	3.6293e+06	23907
T	7639	22443

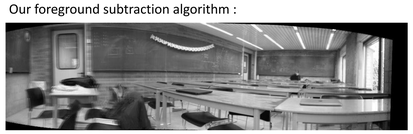
Mean confusion matrix for the KNN algorithm computed on the segments our group annotated.

From the confusion matrix, we can derive the precision and recall metrics.

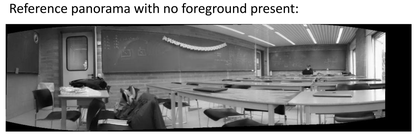
	Precision	Recall
KNN	74%	48%
MOG2	66%	19%

No foreground subtraction:

Our foreground subtraction algorithm :



Reference panorama with no foreground present:



2) Person/ball detection module

To access the performance of the person/ball detection module, we choose to define the detection task as a classification problem with 3 classes: Person, Ball, None. The None class correspond to the case where we have no prediction or no ground truth. More precisely, to compute the confusion matrix for the object detection in a given image, we follow the following methodology:
1. We extracts the list of ground truth bounding boxes and the list of predicted bounding boxes from the image.
2. We compute the IoU of each ground truth bounding boxes with every predicted ones.
3. We keep the best matches according to the IoU measure (to avoid duplicate match).
4. We remove the matches with IoU < 0.5 (threshold). They are considered as not valid.
5 We compute the 8 quantities of the confusion matrix.

The 8 quantities in the confusion matrix can be described as follow:
M11: Ball that is predicted as ball
M12: Person that is predicted as ball
M13: Predicted ball at a location with no object
M21: Ball that is predicted as person
M22: Person that is predicted as person
M23: Predicted person at a location with no object
M31: Ball that is not predicted
M32: Person that is not predicted

The results we get by concatenating the confusion matrices of each images of our test-set are the following:

Pred\True	Ball	Person	None
Ball	41	0	1
Person	1	75	0
None	6	0	0

IoU mean: 0.84493
IoU var: 0.00842

We can see that the main type of error is a ball that is not predicted by the model. As we used it to compute foreground mask in background subtraction, this means the balls were also missing from the foreground masks

3) Qualitative analysis of our algorithm:

As it can be seen on the figures on the left, having no foreground subtraction yields many undesirable effects. The background is occluded and moving entities can be seen several times and are often distorted. Our algorithm effectively eliminates every foreground object in the panorama. However, the resulting panorama presents some artefacts. As it can be seen on the leftmost part of the image, edges are smeared and blurry. This is due small inaccuracies in the rotation matrix used to compute the homographies. Nevertheless, it is important to note that some gitter can also be seen in the reference panorama that was stitched with OpenCV’s high leve stitching API. This shows that even advanced techniques struggle to find the perfect match between two images. To improve this, better acquisition methods that limit vertical tilt like a tripod should be used.