POLYTECHNIQUE MONTRÉAL

UNIVERSITÉ D'INGÉNIERIE

INF6804

Vision par ordinateur
Rapport

TP2 - Segmentation d'objets vidéo

Élèves :
Marc Zhang 2312403
Pierre Chan Kan Leong 2225665

Enseignant :
Guillaume-Alexandre Bilodeau

18 mars 2024

Table des matières

1 Presentation of the two methods

1.1 Background subtraction

Background subtraction performs a two-class segmentation, separating what is considered as the background, where nothing is moving, from the foreground, where objects are in motion. The detection of moving objects is achieved by comparing the foreground with the background. The parts of the image where the absolute value of the difference exceeds a threshold indicate where an object is moving.

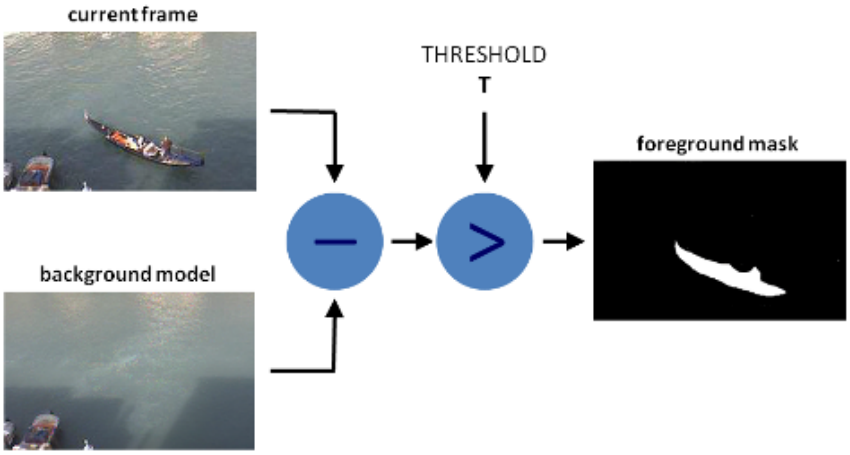

FIGURE 1 – Steps of a background subtraction

1.2 Mask-R-CNN

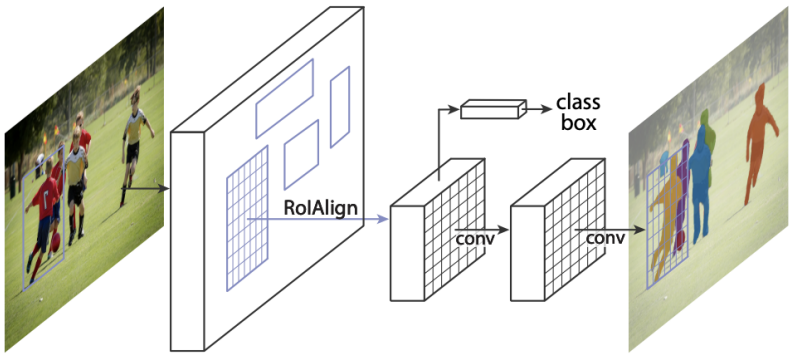

FIGURE 2 – Architecture of Mask-R-CNN

Mask R-CNN perform instance segmentation, each of the predicted mask are associated with an instance of detected object so it can distinguish two different instance of the same class. A high level overview of the method :

— Mask R-CNN [1] start with a backbone network (a pre-trained CNN, we used res-net 50 pretrained on the coco dataset) to extract image features.Then a region proposal

1. https://arxiv.org/abs/1703.06870

network (RPN) proposes candidate object bounding boxes (region of interest). The number of region of interest is a hyperparameter and each box is associated with a score between 0 and 1 which can be interpreted as a probability of containing object of interest.
— Instead of using RoIPool, Mask R-CNN uses RoIAlign to extract features from each RoI. RoIAlign is more precise because it avoids the quantization used in RoIPool.
— We then use the output from RoiAlign to predict two things in parrallel for each RoI, the class/bounding box and the mask. This way the mask does not depend on the class prediction. The prediction mask has the same dimension as RoIAlign output and contain value between 0 and 1 that can be interpreted as the probability of this pixel being an object of interest, it is resized and re-positioned to the original image size using bi-linear interpolation.

To handle any image dimension, the backbone first resize the image to a [256, 480] dimension and a 224×224 crop is randomly sampled. The hyperparameters we choosed to control was the score threshold of the region proposal network (RPN), and a mask threshold to transform the probability mask prediction to a binary mask prediction.

Using masked R-cnn we need to filter the object of interest, for instance we want to only detect the car in the highway folder, the person and bike in the pedestrians folder ...

FIGURE 3 – Highway mask prediction

1.3 Tools used

To perform the background subtraction, we will use the BackgroundSubtractorMOG2 class from OpenCV [2]. For the mast R-CNN, we used the resnet50 fpn model from the torchvision models detection module [3]

To change the brightness of images, we will use the ImageEnhance module of the Pillow library [4] and to blur the images, we used the GaussianBlur module from the OpenCV library. [5]

2. https://docs.opencv.org/4.x/de/de1/group__video__motion.html
3. https://pytorch.org/vision/main/models/generated/torchvision.models.detection.fasterrcnn_resnet50_fpn_v2.html
4. https://pillow.readthedocs.io/en/stable/reference/ImageEnhance.html
5. https://docs.opencv.org/4.x/d4/d13/tutorial_py_filtering.html

2 Performance hypotheses in specific use cases

2.1 Partially observed

FIGURE 4 – Partially oberved bike

When an object is partially visible, for instance when a person on a bicycle enter the frame of the camera we first see part of the wheel on the the frame then the entire wheel and finally the person on the bike, because CNN does not work on a pixel by pixel level but use filter that first detect smaller features like edges in the first layer and than more general filter like 'human' filter on the last layer, the lack of context (figure 2 we only half a circle to predict a bike) may lead to poor performances. However, the performance of background subtraction is not impacted by the shape or the type of the object, as this method does not consider these aspects.

2.2 Blurred images

The degree of blur can impact the performance for the Mask R-CNN, if the general shape and features are still distinguishable CNN should generalised well however when the contour of object of interest become blurry the predicted mask will be less precise at the border of the mask but should still be accurate for the inside part of the object. If the model have been trained with augmented data including blurred image than it will be robust to blur. The threshold to select region of interest should be lowered as the blurring effect will make the model more uncertain of his prediction. The degree of blur can also impact the performance of the background subtraction, the blur can add noise that can create a difference between the foreground and the background that can be detected.

2.3 Brightness

The brightness of the image will impact the performance of the background subtraction when it tends to extreme values. When the brightness is too high or too low, the contrast between the background and the moving foreground is also reduced, leading to higher chances for the movement to not be detected.

3 Description of experiments, datasets and evaluation criteria

3.1 Evaluation criteria

For each experiment we used the union over intersection of the ground truth and the prediction mask. This metric penalizes false positive (predicted mask with no correspon-

dence with ground truth) and false negative (missing predicted mask), it is invariant to the scale of the images and has easy interpretation with value between 0 and 1.

3.2 Hypothesis evaluation description

For partially observed experiment, we selected part of the video when an object is entering the frame and created a subset of the original dataset and compute the Iou over it.

The gaussian noise is computed with a convolution between the input image and the gaussian kernel. The gaussian kernel is :

$$K(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

where x and y are the coordiante of the kernel, σ is the standard deviation of the Gaussian distribution, controlling the degree of blur.

FIGURE 5 – No blur left, (5x5), (15x15) ,(35x35) kernel size with sigma = 1

And for the illumination Images with different brightness level are computed by multiplying the value of their color by a factor [6]. When this factor is greater than 1, it increases the brightness, and when it is less than 1, it decreases it.

FIGURE 6 – same image with different enhancement values (0.25, 0.5, 2, 4)

3.3 Datasets difficulties

With the Neural net, the ground truth only consider the moving object, however if there is a background object like a bike that is not moving it will be detected by the neural net but count as false positive (see images)

6. https ://pillow.readthedocs.io/en/stable/reference/ImageEnhance.html

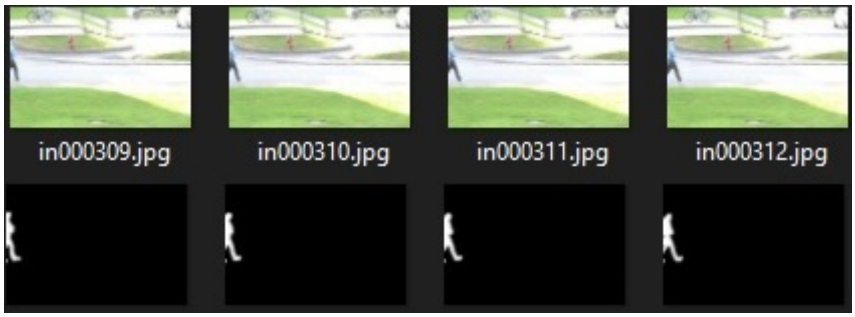

FIGURE 7 – Not moving bike on the background

Some object like shadows are considered in the ground truth but the neural net was not trained to recognize this class

FIGURE 8 – Shadows in the ground truth

The difficulty in subtraction background performance comes mainly when the object of interest moves little and has uniform colors, this is very noticeable in the dataset office where the person is dressed in red and weed and moves little at times.

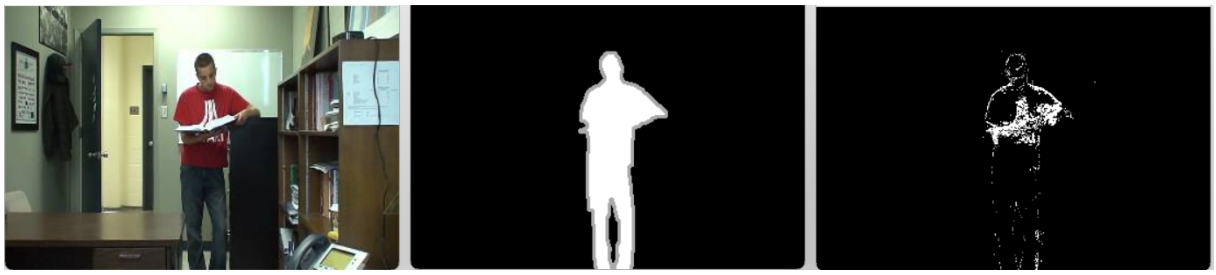

FIGURE 9 – input, ground truth, and output of background subtraction on office dataset

Inversely, where background subtraction performed best was on the dataset highway, where natural lighting that was reflected on cars that the colors on a car were never uniform.

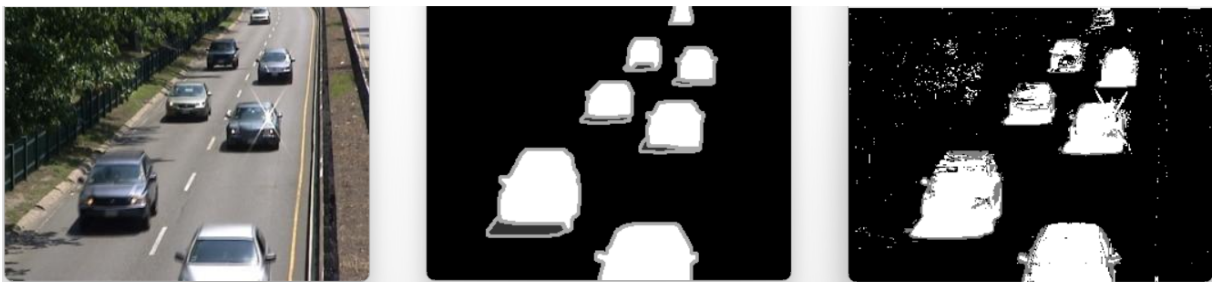

FIGURE 10 – input, ground truth, and output of background subtraction on highway dataset

4 Description of the implementations used

4.1 Masked R-CNN

We used the mask rcnn resnet50 fpn model from the torchvision models detection module. We transformed the images to tensor and feed them directly to the model (the model took care of resizing, center and normalize each image), no other prepossess was done. The output of the model for 1 image is a dictionnary with the key :

- boxes : containing the 4 coordinates of the bounding box
- labels : containing the predicted class of each region of interest
- scores : containing the probability of each proposed region to really contain object of interest
- masks : containing masks of size (224x244) where each pixel represent the probability of being part of an object of interest

We added filter function to only keep the prediction with a high score, and the prediction of object of interest (only car for the highway folder...) and mask threshold function to create binary mask (a mask pixel is considered 0 below the threshold, 1 otherwise)

Hyper-parameters search : we have the score threshold and the mask threshold to optimize, because the forward pass takes a lot of time (the back bone network is a resnet50) we sampled 100 exemples randomly from each folder (highway, office...) and did a grid search.

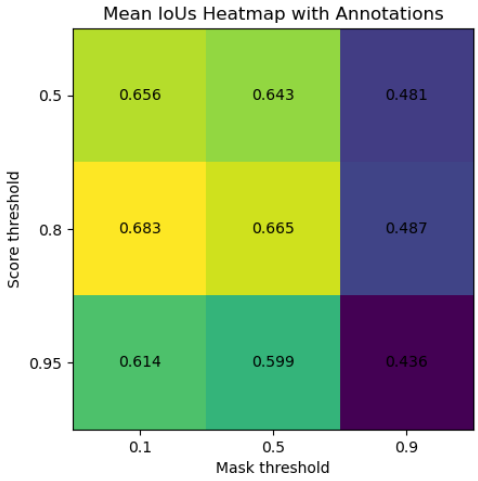

FIGURE 11 – Mean Iou over the 4 folder

The best hyper-parameters combination depends on the folder, smaller object like in the pedestrians folder require a lower score threshold.

	highway	Office	Pedestrians	PETS2006
Best Iou	0.806	0.843	0.488	0.699
(score,mask threshold)	(0.5, 0.1)	(0.8, 0.5)	(0.5, 0.1)	(0.5,0.1)

4.2 Background subtraction

We used the Background subtract from OpenCV, by imputing an image it gives us directly as a result the foreground mask that we can directly compare to the ground truth. To find the best hyper parameters, we plotted the intersection over union as a function of the threshold.

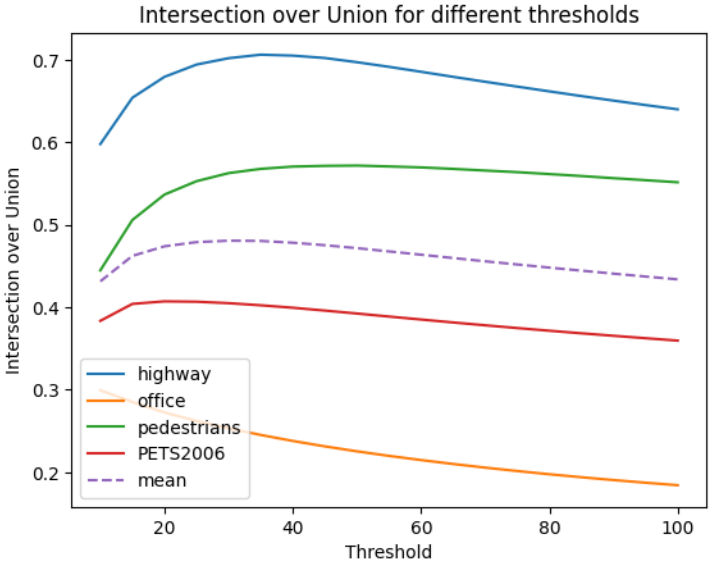

FIGURE 12 – Iou for differents threshold

the general trend is that as the threshold increases, the IOU increases until it reaches a maximum, and then decreases. For future measurements, we will use threshold = 20.

	highway	office	pedestrians	PETS2006'
Iou	0.575	0.326	0.579	0.545
Threshold	15	25	15	25

TABLE 1 – Iou for the best mean value of threshold

	highway	office	pedestrians	PETS2006'
Iou	0.679	0.272	0.536	0.407

TABLE 2 – Iou with threshold = 20

5 Experimentation results, Discussion on results and prior hypotheses

5.1 Blurred images

5.1.1 Neural net

For the neural net we sampled 50 images randomly and gradually blurred them to see the difference in performance of the model. A kernel size of 0 means no blur, the model is robust to small blur but is still able to detect that there is object of interest.

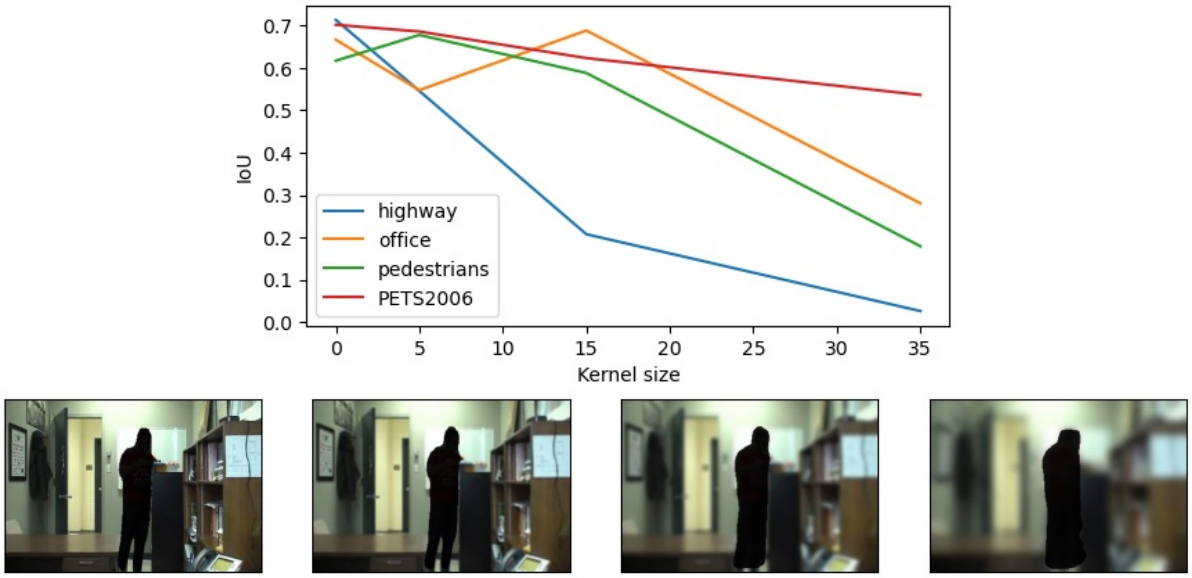

FIGURE 13 – Prediction mask of blurrier version of the same image

We can see that the prediction mask border becomes less precise the blurrier the image which is coherent with our hypothesis. The pixels in the enter of the mask are not affected when the object is still detected. A second type of error is when the object is no more detected because the blurred image removed to much of characteristic feature of the object.

5.1.2 Background subtraction

Since background subtraction takes much less time to execute, we'll perform it on all whole datasets with different kernel size.

FIGURE 14 – Iou for differents kernel sizes

As expected, the background subtraction becomes less precise as the kernel size increases, and we even notice a linear relationship between the two.

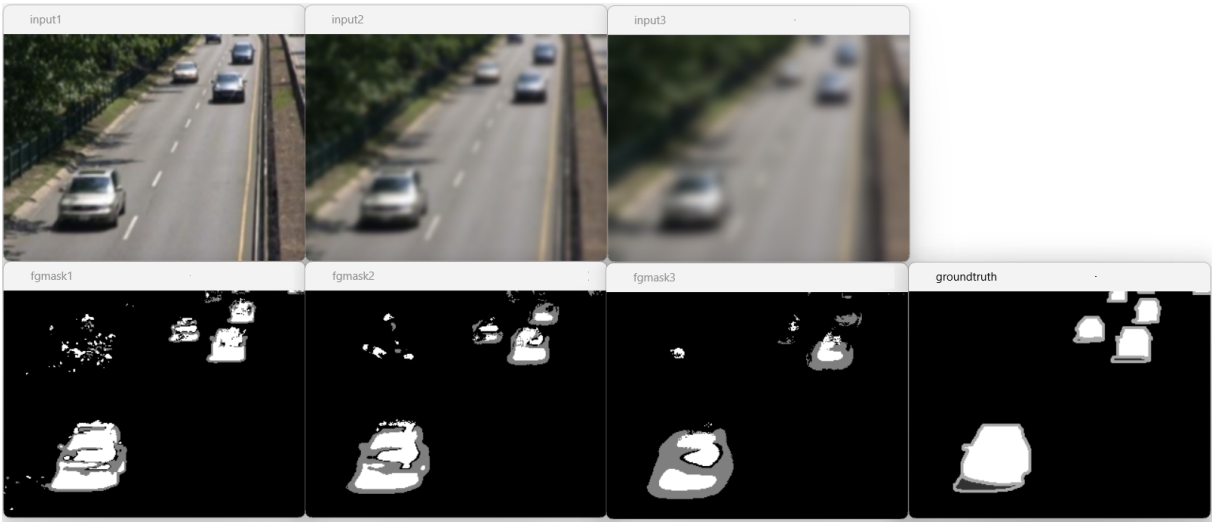

FIGURE 15 – Input, output, and groundtruth for differents kernel sizes

5.2 Partially observed

5.2.1 Neural net

We selected part of the video when an object is entering or leaving the frame of the video. The Iou can be high even if all the object are not detected if other object are easily detectable and much bigger than the partially undetected object.

	highway (600-650)	Office(582-600)	Pedestrians(305-315)	PETS2006 (600-650)
Iou	0.77	0.54	0.58	0.82

TABLE 3 – Iou for partially observed images

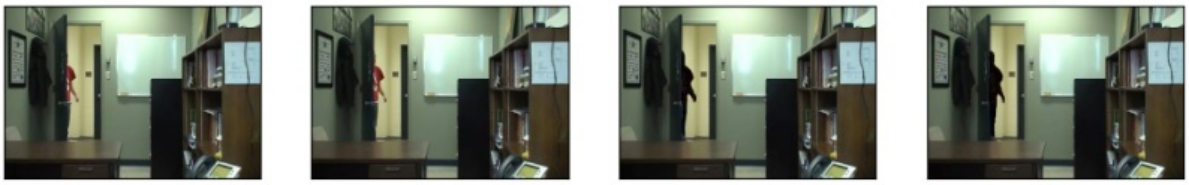

FIGURE 16 – Prediction mask on partially observable human

Qualitatively, we observed that without enough context the model is not able to distinguish the background and the object of interest, the model may have constructed a model of what is a human for instance it has a head, a body and 4 limbs so he is not able to reason that he can only see the hand because the rest of the body is hidden behind the door.

5.2.2 Background subtraction

We've also applied background subtraction to extracts where objects are partially shown, and here are the results :

	highway (600-650)	Office(582-600)	Pedestrians(305-315)	PETS2006 (600-650)
Iou	0.502	0.483	0.499	0.404

TABLE 4 – Iou for partially observed images

No general trend can be observed compared to the application of the method to the whole dataset. We do, however, have a decrease in accuracy for the highway dataset and an increase in accuracy for the office dataset, but given that partially mounting an object does not influence the results of the method, these fluctuations are due to low sampling.

5.3 Illumination

5.3.1 Neural net

A light factor of 1 is the original image, a factor > 1 means the image is brighter and < 1 means the image is darker. A high factor lead to the high loss in performance because the shape of the object of interest are no longer distinguishable (see figure 10) although a human could still draw binary mask knowing the context : the object are on a road so if one can see a moving object it is likely a car.

Maybe a vision transformer model could use this context information with the attention mechanism and perform better on this case.

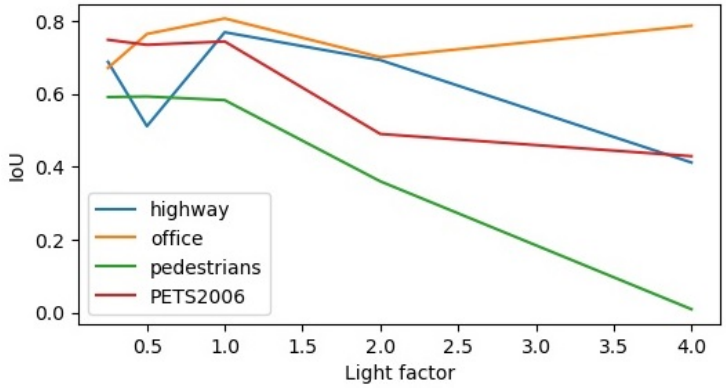

FIGURE 17 – Iou for different light factor setting

FIGURE 18 – Prediction mask for different illumination setting

5.3.2 Background subtration

We measured the Iou on the various datasets for different brightness factors :

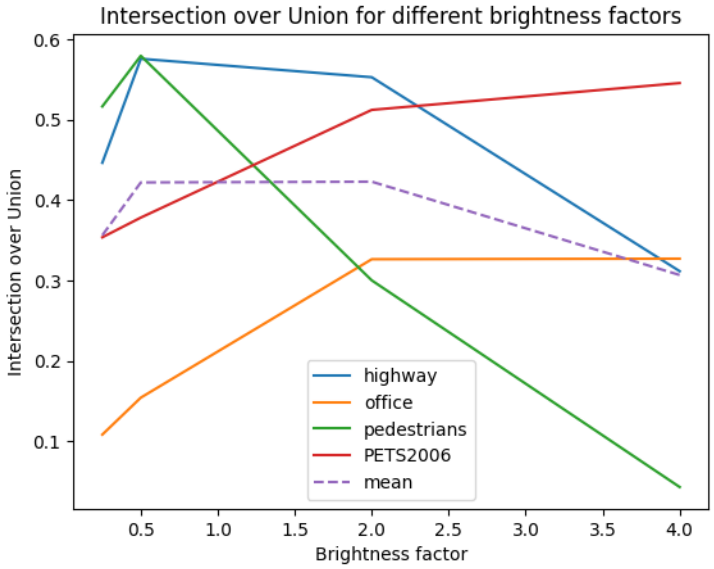

FIGURE 19 – Iou for differents brightness factors

As we predicted, the general trend shows that the closer we get to extreme values, the more our accuracy drops. However, we note that the drop is much greater when brightness increases than when it decreases, and we even notice that with a brightness factor of 0.5 it increases.

FIGURE 20 – Input, output, and groundtruth for differents brightness levels

We can see here that it's as we thought the loss of contrast that leads to a loss of performance at extreme brightness values, as the method no longer detects motion.

5.4 Comparison of the two methods

In terms of performance, the general trend is that CNN performs better than background subtraction. However, this can vary greatly depending on a number of factors. For example, whether you're looking for a stationary object or a moving one. A stationary object will be detected by the CNN, whereas this is not the case for background subtraction, and vice versa for a moving object. Contrast in the object of interest also plays a very important role, as background subtraction can lose a great deal of performance when the object has uniform colors.

In terms of robustness to the perturbations tested, we have roughly the same performance variations for object obstruction and for adding blur to the image, with little difference when the object is obstructed and a decrease in performance when the image is blurred. However, for brightness variation, we have close responses when brightness increases, but much better robustness for CNN when brightness decreases.

In terms of calculation speed, background subtraction is much faster than calculation with a CNN. Although this depends very much on the characteristics of the system used for calculation, we note a calculation time of around twenty seconds for background subtraction for an entire dataset and a calculation time of 10 minutes for 50 images with a batch of 8 and without acceleration using a GPU.