***Abstract*—Pedestrian detection targets to classify and localize multiple people and return the spatial location of each person from images and videos. Though after 30 years of research, many works achieved remarkable results on object detection, most of detectors are not ideal for pedestrian detection in crowded scenes. One reason is that the majority of training datasets only label the uncovered parts of objects which make occluded features hard to be learned. Moreover, most monocular detectors do not take into consideration the multi-camera geometry to resolve ambiguities and ignore occlusions. In this work, we present a model to apply multi-view detection based on monocular instance segmentation using CNNs and multi-view geometry to localize, which is intended to detect people in crowded scenes. This report shows the usage of pedestrian detection, the challenge for detection, the proposed methodology, and its performance on the terrace video sequences.**

***Index Terms*—multi-pedestrian; multi-view; RCNN;**

I. Introduction

PEDESTRIAN detection targets to classify and localize multiple people from images and videos, which is a long-standing problem in computer vision. It has a vast range of applications which can be directly applied in the field of real-world tasks, such as vehicle-assisted driving system, intelligent video surveillance and robots.

Vision-based pedestrian detection technology has been developed over decades, most of them rely on learning algorithms to find the implicit representation of people. There are two difficulties in the design of pedestrian detection system of finding precise features of pedestrians.

1) Inconstant appearance: Pedestrians' appearance varies because of the change in height, pose, sex, clothing and accessory.

2) Occlusion: Pedestrians are moving objects who may be occluded by other objects, especially the occlusion among pedestrians in crowded scenes.

Detecting pedestrians through multiple cameras within an overlapping field of view, allowing to merge more spatial information, is a reasonable solution for occlusion. This work use homography transformation to fuse information of potential location of pedestrians in a top view to indicate their positions in all camera views

This work's contribution is: a simple method to apply current state-of-art deep learning detection technology in multi-view detection which is pre-trained by a monocular dataset and provide a more robust detection compared with methods using background subtraction, HOG and SVM.

II. Related work

The traditional paradigms before 2010 were based on variants of feature extraction techniques: HOG (Histograms of Oriented Gradients) and DPM (Deformable Parts Model). HOG extracts features relied on the edge and shape, like color contrast between pedestrians and background, so it is not effectively record the visible information of objects and it is sensitive to noise. As regards the other method, DPM is more advanced, but it need to manually define features and it is not robust to detect objects with rotating, stretching, and changing the visual angle.

Monocular detection works based on CNN, such as YOLO series and RCNN series have made remarkable achievement on COCO dateset, since deep convolutional neural networks are highly robust to lights, color, texture and white balance, and normally have richer dimensional representations of features computed over the input. However, most multi-view detection methods still depends on old and outmoded technologies, because CNNs rely heavily on training datasets enabling researchers to apply multi-view training on pedestrian detection, but they do not exist.

III. Method for Merging Monocular Detector

The overview of our method is shown as1, it takes two inputs from a monocular detector: masks(foregrounds of pedestrians) and bounding boxes. The truncated mid-lines of bounding boxes are then projected to a top view, where the pedestrians' location candidates are defined. Next RSS (Repulsive Spatial Sparsity Filtering) find local peaks and reduces the duplicates.

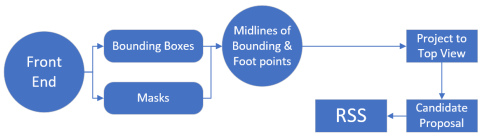

Fig. 1. Method for Merging Monocular Detector

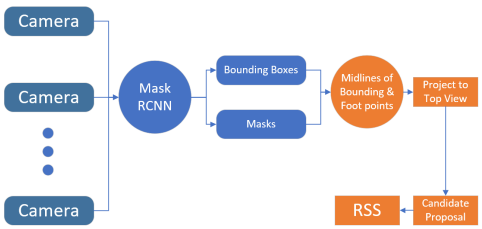

Fig. 2. Overview of the proposed method: using Mask RCNN as front end

A. Candidates Proposal

1) Homography: One difficulty of monocular detection is finding the exact foot positions of pedestrians. The middle bottom point of each bounding box (specifically generated by Mask RCNN in this work) is not the centroid between feet. By default, for eliminating ambiguities, each mid-line of a bounding box is truncated and projected to the top view by homography. A scanner, the horizontal orange line with two gray ends, scans whether it is an occlusion between objects. If either of two ends intersects with the foreground (segmentation), the mid-line should be extended until there is no intersection between the scanner and foreground.

One difficulty of monocular detection is finding the exact foot positions of pedestrians. The middle bottom point of each bounding box (specifically generated by Mask RCNN in this work) is normally not the exact pedestrians' positions. By default, to eliminate ambiguities and find the real positions of pedestrians, each mid-line of a bounding box is truncated, extended and projected to the top view.

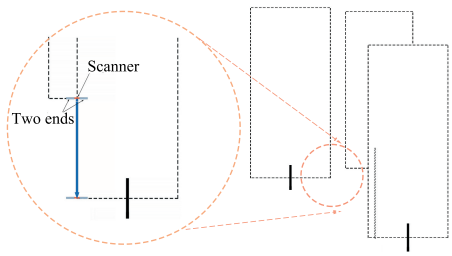

Fig. 3. The truncated mid-lines and masks

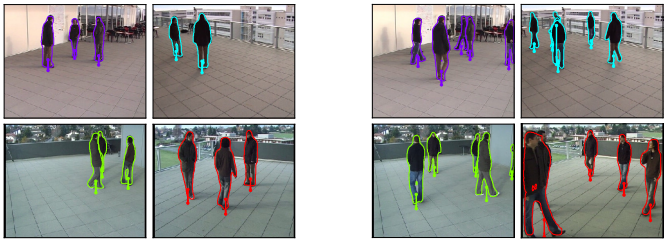

Fig. 4. The truncated mid-lines and masks in Terrace dataset: (a) frame 800th. (b) frame 1775th

A scanner, the horizontal orange line with two gray ends, scans whether it is an occlusion between objects. If either of two ends of a line intersects with the foreground (segmentation), the truncated mid-line will be extended until there is no intersection between the scanner and foreground.

2) Candidates Types: After setting the length of truncation, the mid-lines are projected from all views to a top view, then the next is to propose potential pedestrian's location on ground. There are three types of candidates which locate the potential positions. First type is the intersection of projected lines, it aims to find the ground centroid position of pedestrians. Second type is a set of points between to non-intersected lines, it aims to prevent missed detection while the object is detected by opposite angles. Third type it a set of points on a line who have no neighbors within a specific range, it aims to provide candidates for the person at the edge of overlapping field, where the object is only detected by a signal camera.

B. Multi-view Mask Scoring

Occlusion can become a serious problem in complex scenarios with multiple targets. The score generated by Mask RCNN aims to assess classification results, which is not well correlated with multi-view detection and it can not be used for disambiguate ghost detection.

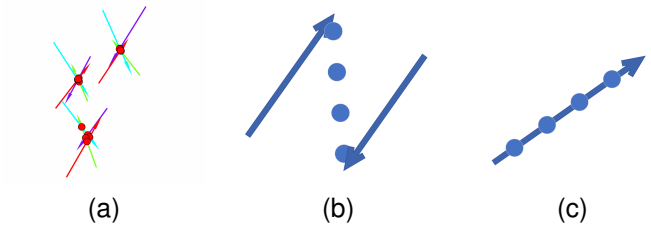

Fig. 5. Three types of Candidates: (a) two lines' intersections. (b) points within parallel lines (c) points on non-intersected line

A likelihood function is applying to candidates after inverse projecting them to camera views and generating numbers of N (N stands for the number of camera views) corresponding bounding boxes based on their positions on the top view. It is assumed the width of a bounding box is 37.5% its height in this step.

The likelihood function is defined as

$$S_{\text{total}} = \sqrt[n]{\prod_{k=1}^{n} (S_{head}[k] * S_{foot}[k])}$$

Where n stands for the number of cameras who detect a same pedestrian by a monocular detector. Since it is hard to know whether bounding boxes frames a same target across multiple views,so that n is counted by the number of valid re-projected bounding boxes (A valid bonding box should be in its corresponding camera's field of view) for a candidate.

1) S_{head}: S_{head} aims to measure the horizontal deviation of a bounding box, whose range is from 0.2 to 1.0.

$$S_{head} = \frac{\sum hist[i] \cdot y[i] + \sum |hist[i] \cdot y[i]|}{2 \sum |hist[i] \cdot y[i]|}$$

2) hist: The masks are binary images whose values are either 1 or -1 (if the original masks values are 1 and 0, change all 0 to -1)

$$mask[i][j] = \begin{cases} 1 \to 1 \\ 0 \to -1 \end{cases}$$

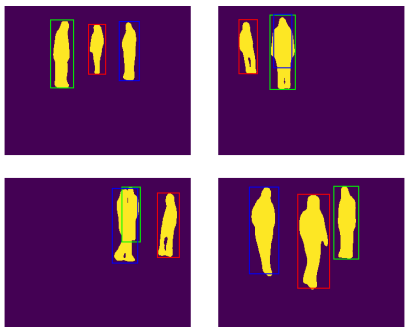

Fig. 6. Masks: 800th frame of terrace dataset, view 1 to view 4

Then count pixel values and add them vertically by every column in the bounding boxes generated from candidates, which are points in the top view.

$$hist = \left[\sum_{i=b}^{t} mask[i][0] \ldots \sum_{i=b}^{t} mask[i][col] \right]$$

Where b is the bottom raw of bounding box and t is the top raw of it.

The drawback of S_{head} is that , when faced with occlusion problem, the bounding box often has higher response in the middle of two targets, because there are more foreground pixels, so that $hist$ has larger positive value, but . Therefore,

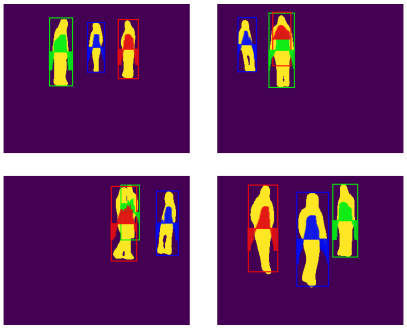

Fig. 7. Histogram of mask column by column: A ideal bounding box should have positive values in the middle and negative on both sides

$y[i]$ in (III-B1) should twist $hist$ values to correct this problem. An ideal bounding box is supposed to be located at the center of target and bound other targets as less as possible.

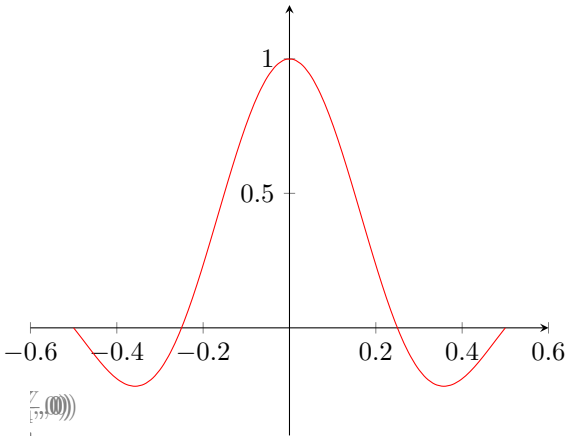

Fig. 8. Mapping Filter Function $y = \frac{sin(4\pi x)}{4\pi x}$

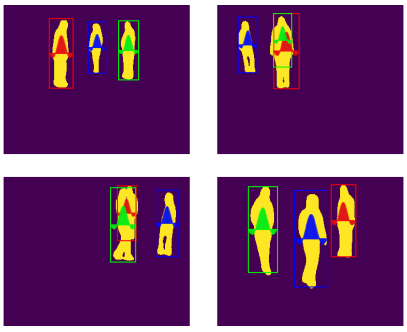

Fig. 9. Mapping functions $y[i]$ is shown in bounding boxes:

3) S_{foot}: S_{foot} aims to measure the vertical deviation at foot. The cumulative distribution function of a normal distribution is used to measure how far is it from foot to bottom of the respective bounding box. Also, to punish low-confidence candidates generated by the inappropriate truncated length of mid-lines, overlap ratio in one-fifth of the bounding box(lower part) between foreground(segmentation) and bounding box

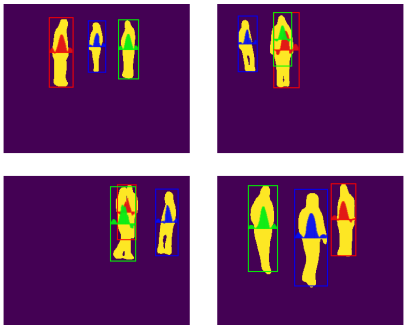

Fig. 10. $hist[i] * y[i]$: Because the three pedestrians are perfectly detected, values are nearly all positive

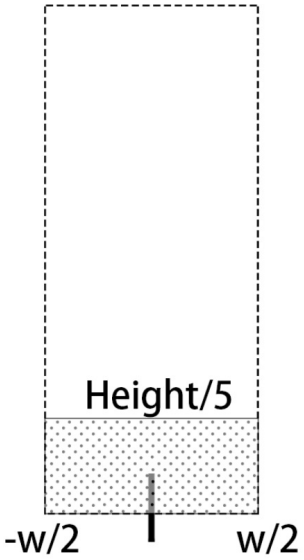

Fig. 11.

area is multiplied with S_{foot}.

$$S_{foot} = 2 * CDF\left(foot - Bottom, \sigma^2\right) * \text{ Overlap Ratio}$$
$$= \frac{2}{\sigma\sqrt{2\pi}} \int_{-\infty}^{\text{foot}-BoxBottom} \exp\left(-\frac{t^2}{2\sigma^2}\right) dt$$
$$* \frac{\sum_{i=-\frac{w}{2}}^{\frac{w}{2}} \sum_{j=bottom}^{\frac{height}{5}} \text{foreground } [i,j]}{w * h}$$

C. Soft Region Non Maximum Suppression

NMS (Non Maximum Suppression) is widely used in object detection for clean up the predictions , but it is also one of reasons that cause difficulties in dense object detection. NMS pick out local maximum recursively and eliminate unnecessary predictions based on IOU of bounding boxes. It is a rash method that make wrong execution.

To enhance the ability for detection in crowded scene, we proposed a non maximum suppression variant that not only use IOU but also take advantage of the geometric information. That is to say, the positions on the top view are helpful to improve NMS performance under occlusion between same class targets.

Algorithm 1 SRNMS

INPUT
K: Radius on the top view
T: NMS thresh
C: Confidence thresh
Z: A set of candidates
OUTPUT
R: A set of results

$R \leftarrow \emptyset$
while $Z \neq \emptyset$ **do**
  sort(Z)
  $R \cup Z[0]$ and $Z - Z[0]$
  for B in Z whose dist($Z[0]$,item)$< K$ **do**
    for $view$ in $views$ **do**
      $B_{it} \leftarrow inverse\ transform(B, view)$
      $Z[0]_{it} \leftarrow inverse\ transform(Z[0], view)$
      $mIOU = max(IOU(B_{it}, Z[0]_{it}, view))$
      if $mIOU > T$ **then**
        $B_{score} = B_{score} * e^{-mIOU^2}/dist/K$
      end if
      $Z \leftarrow Z - Z[0]$
    end for
  end for
end while
for $candidate$ in R **do**
  if $candidate_{score} < C$ **then**
    $R \leftarrow R - candidate$
  end if
end for
return R

IV. Results

Front end	MDR	FDR	TER	Precision	Recall
MaskRCNN (ResNet 101)	0.	0.0194	0.0356	0.9806	0.9838
Yolo v3 + Deeplab v3+	0.0304	0.0305	0.0609	0.9695	0.9696
MaskRCNN (MobileNet v1)	0.0522	0.0326	0.0848	0.9622	0.9478

TABLE I
Using Different Frond Ends to Extract Bounding Boxes and Segmentation

Appendix A
Proof of the First Zonklar Equation

Appendix one text goes here.

Appendix B

Appendix two text goes here.

Acknowledgment

The authors would like to thank...