Risky Region Localization with Point Supervision

Kazuki Kozuka
Stanford University, Panasonic
kazukik@stanford.edu

Juan Carlos Niebles
Stanford University
jniebles@cs.stanford.edu

Abstract

We study the problem of localizing regions in an image that depict potentially risky areas. In particular, we focus on images acquired by a front camera mounted on a car with the goal of localizing image regions where pedestrians are likely to enter the scene suddenly. In this case, we define the risk value at every pixel as the likelihood that a pedestrian will occupy those pixels shortly. This task is very challenging because the risk areas are not easily characterized by appearances of single objects, and therefore these regions exhibit large visual variations. Additionally, the boundaries of the risk regions in the image are not easily defined by human annotators, as they do not tend to correspond to object boundaries. This causes the annotation process to be ambiguous and costly. To overcome the ambiguity in the boundaries of risky regions, we adopt a weakly supervised method for risk region localization and risk value estimation that only requires single point supervision at training time. To evaluate our approach, we augment the Caltech Pedestrian dataset with risk region annotations. Our results show that our weak supervised method outperform fully supervised approaches in risk region localization and risk value estimation.

1. Introduction

One of the key current bottlenecks in self-driving car technology is the ability to perceive the environment in high detail. In particular, this is important because perceiving the objects and events in the surroundings of the car is crucial for navigation and safety. Recent progress in computer vision has achieved impressive performance in some of the perception tasks in images taken from car-mounted cameras: pedestrian and object detection [4, 12, 25, 27] can run in real time without missing objects of various scale, and per-pixel semantic segmentation of the scene [7] can achieve impressive performance in multiple environmental conditaions. Some recent approaches also tackle direct navigation from perception, by attempting to control the steering wheel directly from input images [5]. In spite of all the progress, current self-driving cars still lack the ability to anticipate or predict future events. In particular, the ability to anticipate sudden events such as a pedestrian rushing into the road will be crucial to achieving acceptable safety standards.

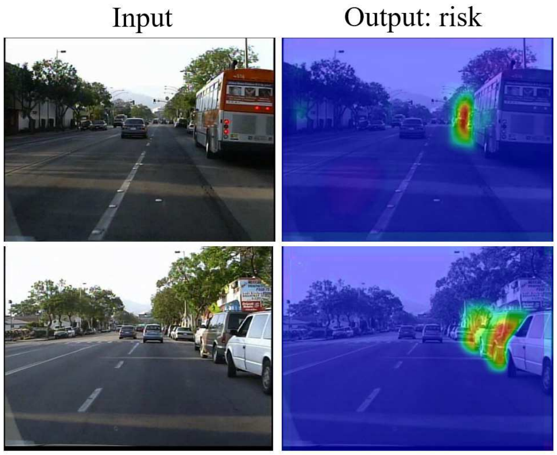
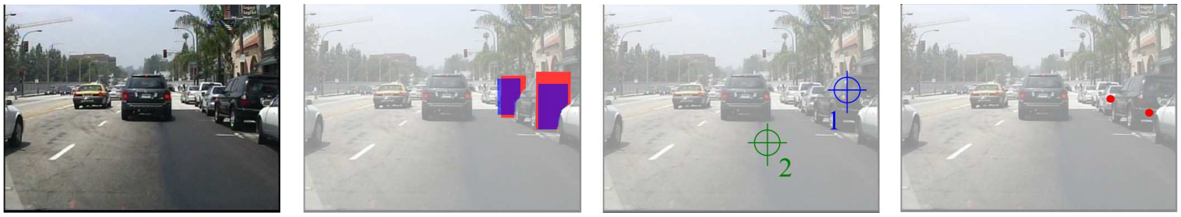


Figure 1. Illustration of risky region localization. left are input images. right are output of proposed method (red area is risky region, blue area is safety region). In the proposed method, risk is predicted not depending on the object categories .

See Figure 1(left) as an example. If a self-driving car relies solely on pedestrian detection algorithms and a pedestrian suddenly rush out from behind the bus, it will be virtually impossible to apply the breaks early enough to avoid a collision. This is because the pedestrian detection algorithm will only respond when the pedestrian is mostly visible. On traditional vehicles operated by human drivers, experienced drivers can anticipate and estimate the risk or danger of the surrounding areas. For instance, if the car approaches areas where pedestrians are likely to rush out, the driver may release the accelerator and get ready to stop ahead of time.

In this paper, we take a step towards providing self-driving cars with such important ability. We introduce a framework to (1) localize regions where pedestrians may rush out and (2) the risk level of such regions. Note that this has to be done even before the pedestrians appear in



(a) original (b) full supervision (c) ordinal supervision (d) 1 point supervision

Figure 2. Levels of supervision for risky region localization. (a) shows an original image. (b) shows full supervision, where the red and blue areas are manually annotated by per-pixel, different colors indicate annotator identity. (c) shows ordinal supervision, where a worker annotates the risk relationship between point 1 and 2, here point 1 is more risky. (d) shows single point supervision adopted in the framework presented in this paper (red points are risky points in the image).

the image. Figure 1(right) shows an example output of the risky region localization and risk level estimation proposed in this paper. For our purposes, we define risk regions as areas where *pedestrians are likely to rush out from behind an object*.

There is not much prior work on the task of estimating risk from visual inputs taken from car-mounted cameras. Jung *et al*. [13] propose a collision warning system via semantic segmentation. They estimate whether the segmented pedestrian regions are dangerous. However, this method only can estimate risk for visible pedestrians. Within the domain of transportation, Shad *et al*. [24] estimate dangerous city areas from historical accident logs but do not rely on any visual data. They achieve this by predicting the risk level from the types of location (intersection, square) and the number of accidents. However, their method operates at a very low spatial resolution and the estimation is independent of the current configuration of cars and other objects in the environment.

While not directly addressing the risk localization problem, some previous computer vision approaches tackle related tasks. The most relevant work focuses on object recognition and semantic segmentation [10, 11, 18, 20, 26]. The usual learning procedure for these methods requires fully supervised training examples, which correspond to object bounding boxes or per-pixel category labels for the entire image. These models then learn the object region or segmentation by minimizing the errors between the prediction result and the ground-truth. One could extrapolate such approach for our task of risk localization. That is, for each training image, we would need to annotate risk region bounding boxes or segmentation masks and use them as ground-truth to train one of those architectures in a fully supervised way. Unfortunately, this strategy is unlikely to be successful. First, object detection methodologies rely on somewhat consistent object appearances. In our case, the appearance varies dramatically, as the risk regions do not correlate directly with object appearances, but instead depend on complex object and scene configurations. On the other hand, semantic segmentation architectures require detailed per-pixel segmentation masks as ground-truth for training. Unfortunately, it is hard to obtain per-pixel masks for the risk localization task. This is in part because the boundaries of the risk regions in the image are not characterized directly by their appearance and they do not correlate well with object boundaries in the scene. This makes the regions difficult to be unambiguously defined by human annotators. Figure 2 (b) shows an example of per-pixel annotation on an image taken from a car-mounted camera. Empirically, we observe that per-pixel risk annotation masks from two workers only have a coincidence of 54% on average (over 3,438 images from the Caltech pedestrian dataset). One way to tackle this issue is avoiding the detailed, costly and ambiguous mask annotation by only collecting single point supervision [3] (Figure 2 (d)). In practice, we observe that single point annotations tend to coincide within a few pixels of each other. While this simplifies the annotation and reduces its costs, it comes with the penalty that no information about the extent of the risk regions is now available at training time. Inspired by [3], we address this by introducing priors appropriate to our problem of risk localization.

In addition to our task of localizing risky regions, we are also interested in producing an estimate of the risk level for each region. Unfortunately, annotating the degree of danger or risk is even more difficult than annotating the location of risky regions. For example, annotation rules are difficult to define because the degree of risk depends on scene arrangement and other factors. Therefore, annotators usually cannot accurately give the numerical value of the risk level of the region. One way to tackle this issue is avoiding risk level annotation by only using pairwise ordinal relationships between points in an image. A similar strategy was adopted by [6] for estimating relative depth values in an image when annotators could not provide accurate depth estimations, but were able to correctly annotate relative depth between pixels. In our case, we can adopt this strategy to automatically provide ordinal relations of risk between pixels as ground-truth. This enables our algorithm to estimate relative risk values from input images. Finally, we incorporate prior information about safe regions in the environment to successfully establish a base risk value of zero in the safe areas.

In summary, our main contributions are: (1) We introduce the first framework for predicting the area where a pedestrian is likely to rush into a road scene. (2) We address the ambiguity in region boundary annotation by single point supervision and propose a method to supplement such weak supervision with prior knowledge.

2. Related Work

Semantic Segmentation. Most semantic segmentation methods [7, 21, 22, 23] require full-supervision which accurately gives the shape of the target objects. In this setting, workers annotate the object areas pixel-by-pixel. However, this setting is difficult to apply directly to our task of risky region localization as the boundary of these areas is not easily determined. Recent work in semantic segmentation has attempted to relax the assumption of fully supervised segmentation masks for training. In particular, Bearman *et al.* [3] propose a method that requires the location of single point within the object region as the annotation. Since a single click is not enough to define the extent of the object, they augment the annotation with the use of an *objectness* prior. The prior improves segmentation performance by providing cues of the extent of the object region. However, note that in our problem of risky region localization, the algorithm should predict the potential risky area before the pedestrian appears. This makes the use of objectess priors less effective in our setting, as the boundaries of the risk area do not coincide with the boundaries of the pedestrian or any other specific object. Inspired by [3], in this work we (1) adopt the idea of single point supervision, where annotators only click one pixel within each risk region; and (2) extend their framework by incorporating priors that are more relevant to the task of risk localization.

Single Image Depth Estimation. Also related to our framework are approaches that estimate a numerical value for each pixel in an image. An example of these are methods for single image depth perception [6, 9, 16, 17]. In particular, Chen *et al.* [6] learn to estimate depth from single images using ordinal information. To train their system, they collect pairwise annotations similar to what is shown in Figure 2(c). In their setting, the annotator selects two points in the image and describes the relative distance of these points with respect to the camera, by indicating which point is closer to the camera or whether both are at the same distance. This circumvents the need for the annotators to provide a specific distance value for each pixel, which would be inaccurate. In our paper, we adopt the idea of pairwise ordinal relationships between pairs of points as a way to train a system that estimates the value of risk at every pixel in the image. Note that, just as in the case of depth, when predicting risky regions, we need to use many pairs having different degrees of risk as training data. Unlike their method that requires annotators to manually click two points in the image, we only ask annotators to click on a single point for each risky region in the image and we automatically generate pairs afterwards. See Section 3.3 for more details.

Trajectory Anticipation. Regarding anticipation to events, recent work has tackled the problem of predicting pedestrian trajectories to estimate where they are likely to go [1, 14, 15, 19]. Kitani et al. [15] propose a method of predicting the trajectory of pedestrians from surrounding physical information. In their method, the model learns from trajectories that pedestrian have taken. For risk region localization, their method cannot be applied because it is difficult to collect large amounts of data that explicitly depict pedestrians rushing out to the road.

3. Risky Region Localization

We are interested in the problem of localizing risky regions in images as well as evaluating their risk level. In particular, we focus our analysis of risk to images captured by a camera mounted on a car. In our setting, we define risky regions as those areas in the images with high likelihood of a pedestrian suddenly appearing. More formally, the input to our framework is an image I as shown in Figure 1 (left), and the output is a pixel-wise risk map z as shown in Figure 1 (right). A key challenge in our problem is that obtaining detailed per-pixel supervision of risk areas as in Figure 2(b) is difficult due to the ambiguous nature of the boundary of the risky regions. Furthermore, per-pixel risk levels are also difficult to estimate by a single human annotator. To circumvent these challenges, we instead ask annotators to simply provide a single point supervision for each risky region in the image, as depicted in Figure 2(d). This makes the annotation process much faster and less costly. We compensate such weak supervision by integrating prior information from automatically obtained safe areas in the image and generating denser supervision in the form of relative risk between a large number of point pairs. In the following, we present the details of our Risk Region Localization Network depicted in Figure 3(top) and our automatic generation of pairwise supervision depicted in Figure 3 (bottom).

3.1. Risk Localization Network

Our framework estimates risk maps z from input images I using a Risk Localization Network. We adopt the hourglass network architecture from Chen *et al.* [6]. In their work, they train a single-image depth estimator from supervision in the form of pairwise ordinal labels. Here, we employ a similar hourglass architecture to generate pixel-wise risk estimates z. We also train our Risk Localization Network using a similar strategy: we provide supervision in the

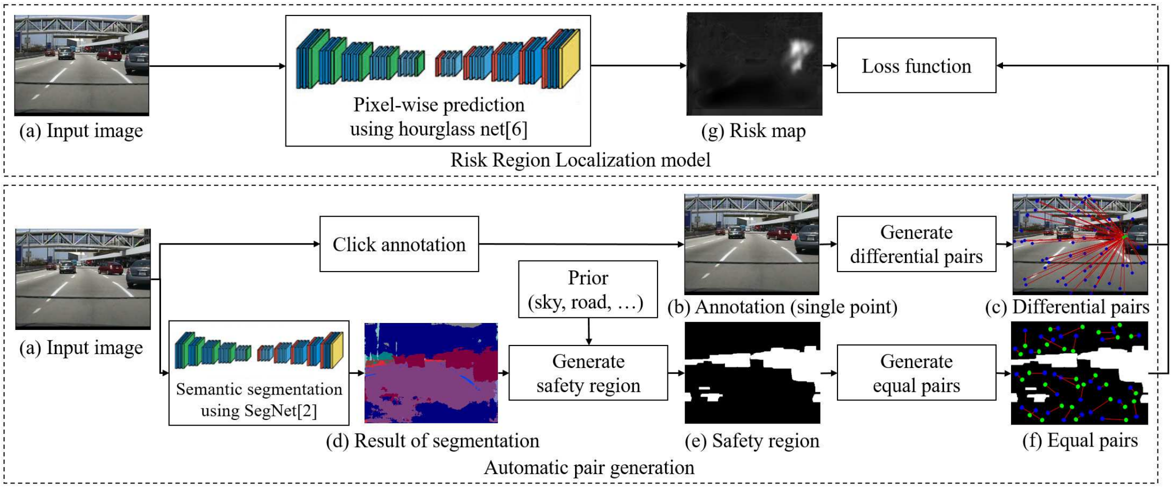


Figure 3. Overview of our risk localization approach. We use single point supervision as annotation, and generate different risk pairs (differential pairs). Also, we utilize prior about safety area by using semantic segmentation, and generate same risk pairs from safety area (equal pairs).

form of pairwise ordinal risk relationships. Note that in [6], supervision is generated manually by annotators that judge ordinal labels for a large number of pixel pairs. This results in large annotation costs, as reflected by the time it takes to annotate each image (see Section 4). We address the annotation cost issue by adopting a semi-automatic strategy for generating pairwise supervision, which includes manual single point annotations and automatic pair generation as detailed in Section 3.3. For each training image I, the result of this process is a large set of K point pairs with annotation $R = (p_{1k}, p_{2k}, r_k), k = 1, ..., K$, where p_{ik} is the location of the i-th point in the k-th pair. By construction, we only consider point pairs where: (1) both points have zero risk, in which case we set $r_k = 0$ (equal risk pair); and (2) p_{1k} is in a risky region and p_{2k} is random point in a image, in which case we set $r_k = 1$ (differential risk pair).

3.2. Loss function

In order to train our Risk Localization Network, we use backpropagation to minimize a differentiable loss that encourages the network to predict relative risk correctly. To accomplish this, we augment the loss function in [6] to incorporate additional constraints that we can leverage in our problem setting. In particular, by construction we only consider equal risk pairs ($r_k = 0$) where the risk value of each point is zero. We leverage this property of our supervised point pairs to extend the original loss function as follows. Let z be the predicted risk map and $z_{p_{1k}}, z_{p_{2k}}$ be the risk at point p_1 and p_2 in the k-th pair. We write our training loss function as:

$$L(I, R, z) = \sum_{k=1}^{K} \psi_k(I, p_{1k}, p_{2k}, r, z) \tag{1}$$

where $\psi_k(I, p_{1k}, p_{2k}, r, z)$ is the loss for the k-th pair

$$\psi_k(I, p_{1k}, p_{2k}, r, z) = \begin{cases} \log(1 + \exp(-z_{p_{1k}} + z_{p_{2k}})) & (r_k = 1) \\ (z_{p_{1k}} - z_{p_{2k}})^2 + z_{p_{1k}}^2 + z_{p_{2k}}^2 & (r_k = 0) \end{cases} \tag{2}$$

Note that we have incorporated additional quadratic terms in Equation (2) to penalize for large predictions of risk values when $r_k = 0$.

In practice, we observe that our augmented loss encourages the model to be more conservative at predicting high risk values, which translates in lower false postitive rates and more stable risk value prediction in consecutive video frames.

We also note that it is key for the training process to have access to both differential pairs and equal pairs. On one hand, a large number of differential pairs is important for the ranking loss in Equation (2) to be effective, on the other hand the equal pairs are useful for setting the base risk value of zero in safe regions.

3.3. Automatic pair generation

The training procedure of our Risk Localization Network requires supervision in the form of pairwise risk relationships R. While we could manually annotate a large number of point pairs with relative risk information, this process is expensive and time-consuming. Instead of such extensive annotation, we only request our annotators to provide a single point for each risky region in the image. This significantly reduces annotation costs at the expense of less detailed supervision. Therefore, we need to convert the set of single point annotations into a large number of point pairs with relative risk information, which is more suitable for our learning process. In the following, we describe our strategy for automatically generating these point pairs. Figure 3(bottom) illustrates this process.

We start the process by asking crowd-sourcing workers to annotate a single point for each risky region observed in the image (Figure 3(b)). Since these points are within risky areas, we know they have a higher risk value than other points in the image. Therefore, we can generate a large number of differential pairs by randomly sampling points in the image that are far from the single point annotations (Figure 3(c)).

As mentioned before, it is important for the training algorithm to have access both to differential and equal pairs. Therefore, we need another strategy to generate pairs of equal risk. We do this by first automatically segmenting the image into semantic regions using the method from [2]. Once we obtain a semantic segmentation, we can use prior information to determine some areas in the image that can be considered as safe. For instance, those regions automatically marked as *sky, road, sign-symbol* and *column-pole* (that is, 4 out of 11 classes in CamVid) cannot be occluding pedestrians. This results in an estimated safe area as shown in Figure 3(e). Finally, we can randomly sample a large number of point pairs within the safe region to generate pairs of points with equal risk (Figure 3(f)). Note that by design all equal pairs also have low risk value. We exploit this to constrain our loss function as described in Section 3.2.

Our strategy effectively incorporates prior information about what regions in the image can be considered as safe. This implicitely defines the maximum extent of the risky regions, by enforcing risk to be low in the safe regions of the image. In contrast to [3], where objectness is used to define the boundary of the objects from the inside of the object regions, our safe region prior helps defining the boundary of the risky regions from the outside.

The process outlined here enables our approach to generate a large number of supervised point pairs from single point annotations. This enables more affordable annotation in comparison with prior annotation approaches. We also emphasize that semantic segmentation is only used to generate training data, so the extra computation cost is not incurred during the inference stage, which only consists of a forward pass of our Risk Localization Network.

4. Dataset and Annotation Cost

In order to evaluate our approach, we have augmented a subset of the Caltech pedestrian dataset from [8] with risky region annotations. The original Caltech pedestrian dataset consists of approximately 10 hours of video taken from a vehicle driving in an urban environment and is annotated with bounding boxes of pedestrians and detailed occlusion labels. For the purpose of our experimental validation, we have further annotated over 4,500 images from the Caltech dataset with risky region labels. As indicated in Table 1, each image in our training and validation subsets was annotated by two crowd-sourcing workers, while all images in the testing subset were annotated by ten workers. As ground-truth for evaluation, we annotated the risky regions to 1,193 test images. Since mask annotations for risky regions can have large individual differences among annotators, we employ ten workers to annotate each testing image. We can aggregate the resulting masks to generate a groundtruth probability risk map. In our evaluation, we binarize the resulting map at 0.5 and use the resulting binary image as groundtruth.

Data	Images	Workers
Train	2,260	2
Validation	1,178	2
Test	1,193	10

Table 1. Caltech risk dataset. Images means the number of images. Workers means the number of annotation workers.

For comparisons, we adopt three annotation strategies: full supervision, ordinal supervision and single point supervision. We elaborate the details of each in the following.

Full supervision (32.2 sec/img). We manually annotated segmentation masks for all risky regions, as in Figure 2(b). We find that it takes 32.2 seconds to annotate one image. Unlike semantic segmentation, the type of target area is one category (risky), and since the number of regions is small for one image, it is possible to annotate it relatively quickly. However, since there is ambiguity at the boundary of the risky region, we improve label reliability by increasing the number of annotators per image, which in turn increases annotation cost significantly.

Ordinal supervision (50.6 sec/img, 1.7 sec/pair) Next, we generated pairs of points for the image and asked annotators to judge which point is more risky, as in Figure 2(c). We find it takes 1.7 seconds to annotate one pair. In this experiment, 30 pairs were generated for one image, so the total time required for labeling one image was 50.6 seconds. The annotation can be done easily because the object of judgment is the comparison of two points, but it is time consuming as workers need to judge each pair sequentially.

Point-level supervision (3.4 sec/img) Finally, we collected single point annotations for each risky area where pedestrians are likely to rush out in the image, as in Figure 2(d). We find that the time required for one image is 3.4 seconds. Since there is no need to input the region boundary, the annotation process is less ambiguous. Moreover, using our strategy in Section 3.3, we can generate many supervised pairs in a less expensive manner when compared to ordinal supervision.

From the above results, regarding ambiguities of annotation, ordinal supervision and our single point supervision are easier to annotate than full supervision. Regarding annotation cost, the annotation cost is about one tenth of that of Full supervision. Compared to ordinal supervision, if three pairs or more are generated for an image, then single point annotation is more effective.

5. Experiments

The goal of our experiments is to validate the ability of our trained approach to localize risky regions as well as to estimate the risk value in new unseen images. We compare our full approach to a number of baselines, as well as to simplified versions of our approach. For fair comparisons, all the network structures in the following are fixed to the hourglass architecture of [6].

Baselines

Supervised segmentation: We annotate full-mask of risk to images. Using the full supervision annotation, we use Cross-entropy as the loss function for semantic segmentation [2].

Ordinal: We annotate ordinal relation of risk between 2 points. Pairs are generated by a method similar to the method [6], and we train the risk region localization net by using the ordinal loss function of [6].

Ablations

Point supervision with differential pairs: As shown in Section 3.3, with single point supervision, only pairs with different risk levels are generated, and optimization is performed by the ordinal loss function of [6].

Point supervision with all pairs: As shown in Section 3.3, we generated differential pairs from single point supervision and equal pairs using safety region. These pairs are optimized by the loss function of [6].

Our method

Single point supervision with differential and equal pairs and risk value loss: As shown in Section 3.3, with single point supervision, we generated differential pairs from single point supervision and equal pairs using safety region. These pairs are optimized with the equation (1) and (2) shown in Section 3.2.

5.1. Evaluation Metrics

Our goal is to quantitatively evaluate two aspects of our approach. First, we evaluate the capability of our model to *localize* risky regions in images. Second, we evaluate the performance of our framework when *estimating risk values*.

We evaluate risk localization by thresholding our risk map output z as well as the risk groundtruth. We then compare these binary maps and compute precision, recall and F1 score to measure the performance of our method in localizing risky regions. In practice, an output region is declared as true positive when it has high overlap with a risky region in the groundtruth map and as a false positive otherwise.

We evaluate our estimation of risk value by computing the root mean square error (RMSE) between z and the ground truth risk map. This metric is small when the estimated values are close the the ground truth risk values. We note that the ground truth risk maps are heavily dominated by areas with low risk values, so we also report RMSE separately for high-risk areas (where risk > 0.5) and low-risk areas (where risk < 0.5).

5.2. Risk Region Localization

We report our results for Risk Localization in Table 2. We note that our method requires the least amount of annotation time, as it only relies on single point annotations for each risky region in the training images. However, we note that it achieves similar recall as the fully supervised segmentation approach and it significantly outperforms all methods in terms of precision. In particular, we associate its high precision to the fact that our single point annotation does not give ambiguous boundary annotation but instead focuses on areas that can be reliably marked as risky. We also note that our method outperforms the ordinal baseline, which we attribute to the fact that we can automatically generate many more supervised point pairs and we are not limited by the small number of annotated pairs that can be obtained manually from annotators. Finally, we see in our ablated approaches that incorporating our loss function as well as differential and equal risk pairs are all contributing to the final performance.

5.3. Risk Value Estimation

Table 2 also summarizes our results on risk value estimation. We note that our method achieves the best overall RMSE when compared to our baselines and ablated models. We note that most models achieve low RMSE overall, and low RMSE on low-risk areas, while the error in high-risk areas is higher. Given that the recall results in risk localization are reasonably high, we attribute this discrepancy to the fact that the models are not capable of capturing the risk boundaries precisely and the risk values decay quickly from the center of the risk region. For a practical application in a self-driving car, the precise extent of the risk region may be less critical, as the navigation planner can be take conservative margins around the detected risk areas.

5.4. Qualitative Results

We present some qualitative examples in Figure 4 and 5. First, we discuss the successful cases in Figure 4. Our method correctly predicts the risk regions where pedestrians are likely to rush out against various background areas. For example, it is possible to predict the risky regions even if the color and shape of the vehicle are greatly different or the appearance behind it is greatly different. As illustrated by the examples in the second column, the supervised seg-

	Annotation	Risk Localization			Risk Value		
Method	Time	Recall	Precision	F1 score	RMSE	RMSE	RMSE
(Supervision)	(Sec/img)	(%)	(%)		(All pixels)	(low-risk)	(high-risk)
Supervised segmentation [2]	*32.2*	*77.3*	29.7	0.429	0.216	0.215	**0.239**
Ordinal [6]	50.6	72.5	27.6	0.400	0.128	0.125	*0.280*
Point sup. with differential pairs	**3.4**	68.5	*38.9*	0.496	*0.085*	*0.075*	0.627
Point sup. with all pairs	**3.4**	75.1	38.3	*0.507*	0.093	0.083	0.599
Our method	**3.4**	**77.4**	**43.7**	**0.558**	**0.072**	**0.061**	0.625

Table 2. Quantitative results of risky region localization, Quantitative results of RMSE. Time means annotation time for each image, and measures recall(detection rate) of risky regions, precision of risky regions, F1 score. We also measure root mean square error (RMSE) of whole data, RMSE that the value of risk < 0.5, and RMSE that the value of risk > 0.5.

mentation method detects shapes close to the annotation as risky regions. However, even if a plurality of regions exists in the detected regions in the image, both are high-risk levels. On the other hand, the model with ordinal supervision (third column) outputs higher risk in the region near the car, and the degree of risk is different for each region. On the other hand, if we use only single point supervision, the relative information is optimized, so the position with high risk matches the other method. However, the degree of risk is high overall in the image, and it cannot be used as a risky region localization for self-driving cars. By adding the prior of the safety area, information on the safety region in the image is added, so the detection performance is greatly improved. By adding value to the loss function, false positive is suppressed, and the performance is improved.

Finally, we illustrate failure examples in Figure 5. We note that the person area on the bicycle is erroneously detected. In this dataset, only the information of the risky regions is given without any knowledge about the type of objects in the scene. Therefore, when vertical edges of a person come out strongly, there are possibility that it is determined as risky regions by mistaking it as a concealed object. On the other hand, we cannot detect the risky regions appearing in the opposite lane while turning the corner by all methods. Danger scenes in the corner are few in the dataset, and the number of samples in the dangerous area is small, so it is considered that undetected has occurred. This could be addressed by focused data augmentation of risky scene configurations. In another example, the area of a car running in parallel is erroneously detected. Since the current model uses only a single image, if there are many dangerous scenes in the relative arrangement, the area is erroneously detected. Therefore, it is necessary to use time-series information for vehicles running in parallel where the relative relationship does not change.

6. Conclusion

We introduce a new risk region localization task and propose a first approach to tackle this problem. We propose a weakly supervised method for risk region localization and risk value estimation that only requires single point supervision at training time. Our method only relies on color data from single images, so further exploration may study the use of video sequences and 3D geometry to further improve the performance.

References

[1] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese. Social lstm: Human trajectory prediction in crowded spaces. In *CVPR*, 2016. 3

[2] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *arXiv:1511.00561*, 2015. 5, 6, 7

[3] A. Bearman, O. Russakovsky, V. Ferrari, and L. Fei-Fei. What's the Point: Semantic Segmentation with Point Supervision. In *ECCV*, 2016. 2, 3, 5

[4] Z. Cai, Q. Fan, R. Feris, and N. Vasconcelos. A unified multi-scale deep convolutional neural network for fast object detection. In *ECCV*, 2016. 1

[5] C. Chen, A. Seff, A. Kornhauser, and J. Xiao. Deepdriving: Learning affordance for direct perception in autonomous driving. In *ICCV*, 2015. 1

[6] W. Chen, Z. Fu, D. Yang, and J. Deng. Single-image depth perception in the wild. In *NIPS*, 2016. 2, 3, 4, 6, 7

[7] Y. Chen, D. Dai, J. Pont-Tuset, and L. V. Gool. Scale-Aware Alignment of Hierarchical Image Segmentation. In *CVPR*, 2016. 1, 3

[8] P. Dollár, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: An evaluation of the state of the art. In *PAMI*, 2012. 5

[9] R. Garg, B. V. Kumar, G. Carneiro, and I. Reid. Unsupervised cnn for single view depth estimation: Geometry to the rescue. In *ECCV*, 2016. 3

[10] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 2

[11] H.H.Clark. Coordinating with each other in a material world. In *Discourse Studies*, 2005. 2

[12] J. Hosang, M. Omran, R. Benenson, and B. Schiele. Taking a deeper look at pedestrians. In *CVPR*, 2015. 1

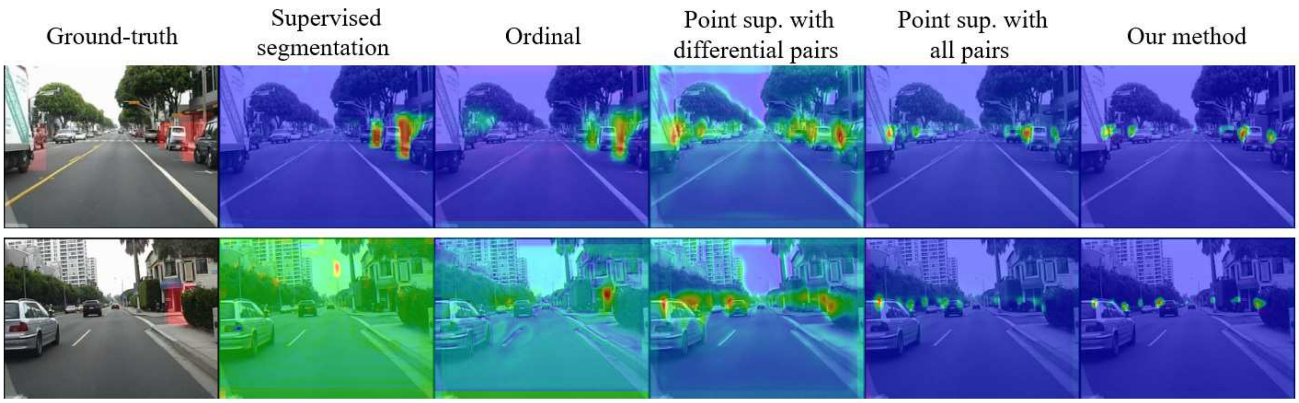


Figure 4. Qualitative Results (success examples). 1st column shows annotation. 2nd column shows the result of supervised segmentation. 3rd column shows the result of ordinal supervision. 4th column shows the result of single point supervision with differential pairs. 5th column shows the result of single point supervision with all pairs. 6th column shows the result of our method.

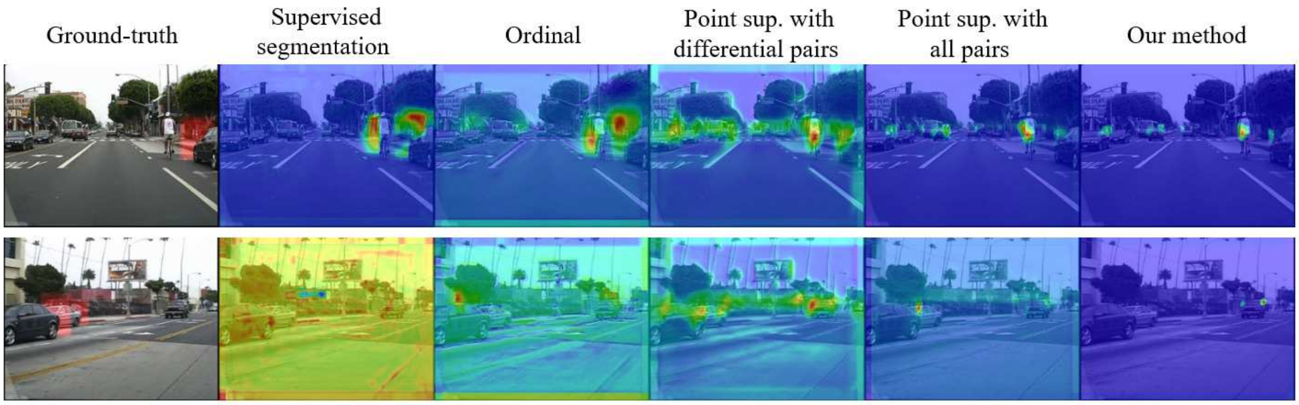


Figure 5. Qualitative Results (failure examples). Columns are aligned with Figure 4.

[13] H. Jung, M. Choi, S. Kwon, and W. Y. Jung. End-to-end pedestrian collision warning system based on a convolutional neural network with semantic segmentation. *arXiv:1612.06558*, 2016. 2

[14] V. Karasev, A. Ayvaci, B. Heisele, and S. Soatto. Intent-aware long-term prediction of pedestrian motion. In *ICRA*, 2016. 3

[15] K. M. Kitani, B. D. Ziebart, J. A. D. Bagnell, and M. Hebert . Activity forecasting. In *ECCV*, 2012. 3

[16] J. Li, R. Klein, and A. Yao. Learning fine-scaled depth maps from single RGB images. *arXiv:1607.00730*, 2016. 3

[17] F. Liu, C. Shen, and G. Lin. Deep convolutional neural fields for depth estimation from a single image. In *CVPR*, 2015. 3

[18] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 2

[19] W. Ma, D. Huang, N. Lee, and K. M. Kitani. A game-theoretic approach to multi-pedestrian activity forecasting. *arXiv:1604.01431*, 2016. 3

[20] G. Papandreou, L. Chen, K. P. Murphy, and A. L. Yuille. Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation. In *ICCV*, 2015. 2

[21] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello. Enet: A deep neural network architecture for real-time semantic segmentation. *arXiv:1606.02147*, 2016. 3

[22] T. Pohlen, A. Hermans, M. Mathias, and B. Leibe. Full-resolution residual networks for semantic segmentation in street scenes. *arXiv:1611.08323*, 2016. 3

[23] G. Ros, S. Stent, P. F. Alcantarilla, and T. Watanabe. Training constrained deconvolutional networks for road scene semantic segmentation. *arXiv:1604.01545*, 2016. 3

[24] R. Shad, A. Mesgar, and R. Moghimi. Extraction of accidents prediction maps modeling hot spots in geospatial information system. In *ISPRS*, 2013. 2

[25] Y. Tian, P. Luo, X. Wang, and X. Tang. Deep learning strong parts for pedestrian detection. In *ICCV*, 2015. 1

[26] T. Wang, B. Han, and J. Collomosse. Touchcut: Fast image and video segmentation using single-touch interaction. In *CVIU*, 2014. 2

[27] L. Zhang, L. Lin, X. Liang, and K. He. Is Faster R-CNN Doing Well for Pedestrian Detection? *arXiv:1607.07032*, 2016. 1