Auxiliary Tasks To Improve Trip Hazard Affordance Detection On Construction Sites

Sean McMahon[1*†], Tong Shen[2], Niko Sünderhauf[1], Ian Reid[2], Chunhua Shen[2] and Michael Milford[1‡]

[1] Queensland University of Technology (QUT), Australia

[2] The University of Adelaide, Australia

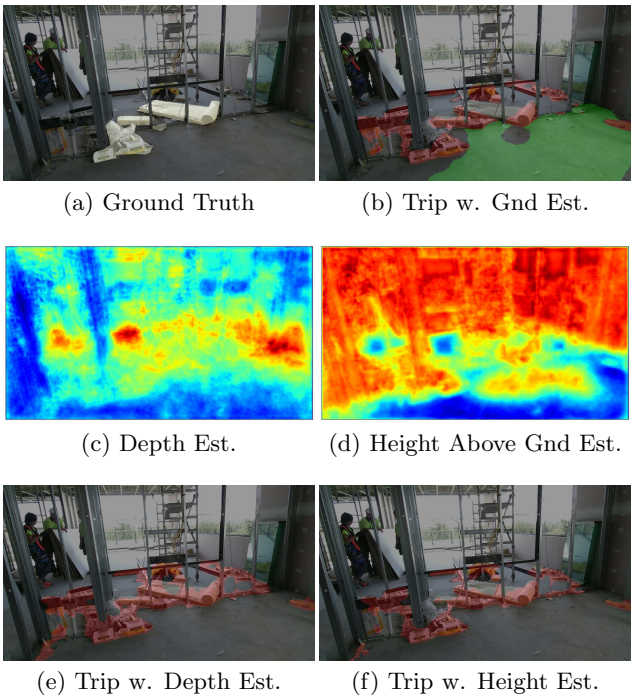
Abstract

Trip hazards are a danger present on all construction sites and failure to identify them can lead to serious injury. The major challenge of detecting these trips is the large number of objects capable of becoming hazards. This challenge is exacerbated because these hazards are not entirely defined by their object type, for example tools on the ground are trippable but not when hung on a wall. We propose to train a CNN performing pixel-wise trip detection with three auxiliary tasks to help the CNN better infer scene geometric properties of trip hazards. Of the three approaches investigated pixel-wise ground plane estimation, pixel depth estimation and pixel height above ground plane estimation, the first approach allowed the trip detector to achieve a 11.1% increase in Trip IOU over earlier work. These new approaches make it plausible to deploy a robotic platform to perform trip hazard detection, and so potentially reduce the number of injuries on construction sites.

1 INTRODUCTION

Trip hazards are one of the most common hazards on construction sites, with 2.8 million people injured in the USA in 2015 [US Department of Labor, 2016]. A vision based trip hazard detector could reduce these incidences when placed on an assistive robotic platform to support human workers on construction sites.

Vision based trip hazard detection on a construction sites is not straightforward, especially when the same object, a ladder on the ground can be a trip hazard but not when hung on a wall. Trip hazards are more defined by their affordance than object type. They are primarily objects on the ground within a height of less than 0.5 meters (from the ground). Rather than a traditional object-oriented method, we propose a more targeted approach where our vision system has a number of auxiliary tasks which allow it to better learn the key geometric attributes commonly associated with trip hazards.



(a) Ground Truth (b) Trip w. Gnd Est. (c) Depth Est. (d) Height Above Gnd Est. (e) Trip w. Depth Est. (f) Trip w. Height Est.

Figure 1: Predictions from the three different auxiliary tasks performed, ground plane estimation (b), depth estimation (c and e), and height above ground estimation (d and f). In sub-images b, e and f, red regions represent trip predictions and the green region are ground plane estimations. While in sub-images c and d, red represents higher and blue represents lower estimation values.

*This research was supported by an APA scholarship and by the Australian Research Council Centre of Excellence for Robotic Vision (project number CE140100016).

†Email: sean.mcmahon@roboticvision.org

‡Michael Milford was partially supported by an Australian Research Council Future Fellowship FT140101229

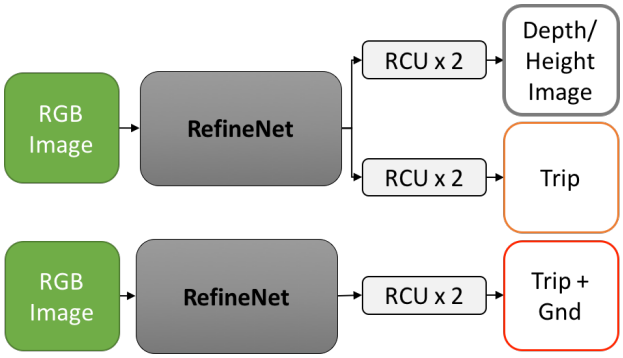


Figure 2: The two frameworks used to investigate the three auxiliary tasks. The top framework shows how the depth and height above ground plane auxiliary tasks are incorporated into the CNN architecture. The bottom framework shows how the ground plane estimation tasks is added to the CNN architecture as an additional class.

This research investigates the effects of three auxiliary tasks on trip hazard detection using a Convolutional Neural Network (CNN), to perform pixel-wise classification.

Three different auxiliary tasks are introduced and investigated, pixel-wise ground plane estimation, estimation of pixel depth, and estimation of pixel height above ground. The state-of-the-art CNN, RefineNet [Lin *et al.*, 2017] is trained as a baseline approach to perform pixel-wise trip detection. To train RefineNet to perform the additional auxiliary tasks, ground plane estimation is undertaken as a classification task, while pixel depth and height above ground are undertaken as regression tasks (see Figure 1 for visualisations). This research differs from earlier attempts in multi-modal fusion [McMahon *et al.*, 2017] by inferring aspects of scene geometry from colour images with the addition of three auxiliary tasks and the updated CNN, RefineNet.

The investigation found the first approach, ground plane estimation, achieved the highest trip detection performance. A 2.8% improvement was shown over our baseline approach and a 11.1% increase over the earlier attempt on multi-modal fusion [McMahon *et al.*, 2017] in the Trip IOU metric. This research is significant as it introduces three different auxiliary tasks, two of which improve the trip detection baseline and establishes ground plane estimation to be the highest performing auxiliary task.

2 Related Work

The related works are shown in three categories, Affordance Detection, Image Processing with Convolutional Neural Networks (CNNs) and Geometry Estimation. As trip hazards are a type of affordance, the first subsection Affordance Detection will discuss visual affordance detection, learning affordances from interaction and learning affordances from observing human interactions. The foundation of our approach is a CNN trained for pixel-wise classification (semantic segmentation), therefore in subsection Image Processing with CNNs, we review CNNs in a number of image processing tasks, including image classification, object detection and semantic segmentation. Finally, in the subsection Depth and Normal Estimation analyses the various depth, surface normal and curvature estimation approaches, and approaches these estimations are used as auxiliary tasks.

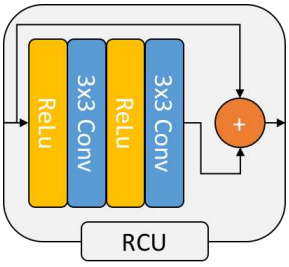


Figure 3: An expanded view of the Residual Convolution Units (RCU) as seen in Figure 2.

2.1 Affordance Detection

The term affordance was first introduced by James Gibson [Gibson, 1978] as a term describing what actions an agent can perform on an object, such as its graspability. The action or affordance space of objects can vary significantly depending on who or what the agent is. Therefore, in this research trip affordances are defined by the affordance space of a human agent. Trip affordance are defined here as an object mostly contained within 0.5 meters from the ground which would cause a person to stumble or fall.

Learning Affordances From Interaction

Affordances are commonly learnt from interaction, which has two main areas, learning from physical interaction and learning from observing human interactions. The first, learning affordances from physical interaction, usually involves some level of physical interaction of the agent, which is usually a robot [Mar *et al.*, 2015; Goncalves *et al.*, 2014; Tikhanoff *et al.*, 2013; Yi *et al.*, 2015; Moldovan *et al.*, 2012]. The second area, involves learning affordances from human observation, where the robot repeatedly observes a set of human interactions [Chan *et al.*, 2015; Gupta *et al.*, 2009; Kjellström *et al.*, 2011; Koppula and Saxena, 2015; Yao *et al.*, 2013b]. The long time duration required to repeatedly interact or observe interactions with objects is one limitation of these approaches. Another is the requirement for physical interaction with these objects,

something which is infeasible for hazards on construction sites, and many other dangerous environments.

Visual Affordance Detection

Another approach for affordance identification is visual affordance detection. There are three categories for these approaches, indirect, semi-direct and direct visual affordance detection. Indirect affordance detection involves manually assigning a set of affordances to a set of objects [Chao *et al.*, 2015; Yao *et al.*, 2013a]. This type of approach is limited by the extent to which the human has assigned affordances to objects. For semi-direct affordance detection object attributes are identified and used a criteria to match objects with their affordances [Hassan and Dharmaratne, 2015; Moldovan and De Raedt, 2014]. Direct affordance detection involves learning affordances end-to-end, directly from image pixels [Fouhey *et al.*, 2015; McMahon *et al.*, 2015; 2017], which is the approach employed here.

2.2 Image Processing with Convolutional Neural Networks

Convolutional Neural Networks (CNNs) have achieved state-of-the-art performance in majority of image processing tasks, including image classification [He *et al.*, 2015], object detection [Ren *et al.*, 2015], semantic segmentation [Lin *et al.*, 2017] and depth estimation [Kendall and Gal, 2017]. This research uses a CNN to perform semantic segmentation or pixel-wise classification, the use of CNNs for this task was first popularised by Long *et al.* and has since seen improvements from others [Liang *et al.*, 2015; Oliveira *et al.*, 2016; Chen *et al.*, 2015; Lin *et al.*, 2017; He and Girshick, 2017].

Depth and Normal Estimation

This section will discuss the different approaches for performing depth estimation from RGB images. Depth and surface normal estimation is a challenging task with relativly few bodies of research [Kendall and Gal, 2017; Eigen and Fergus, 2015; Li *et al.*, 2015], all of which use CNNs with a regression loss function. Kendall and Gal applied Bayesian networks for state-of-the-art results. Eigen and Fergus shows that the same CNN can perform both depth estimation and pixel-wise classification tasks, however they do no attempt to do both simultaneously. There are few who perform estimations of scene depth and normals as an auxiliary task. Mirowski *et al.* work introduces an auxiliary depth estimation task for reinforcement learning, training a neural network to navigate mazes in simulation. The addition of a depth auxiliary task was shown improved performance. Hoffman *et al.* trained a CNN to perform object detection, and showed improved detection rates by adding a second network trained to replicate features from depth images.

3 Approach

This section describes the approaches used to perform pixel-wise trip hazard detection with the three different auxiliary tasks. This section has four main subsections, Network Architecture, Auxiliary Tasks, Training Procedure and Construction Site Dataset. First the subsection Network Architecture describes the architecture of the RefineNet CNN used. The second subsection, Auxiliary Tasks explains the approaches to train RefineNet with the additional auxiliary tasks. The third subsection, Training Procedure describes the visual dataset used, how we evaluated our approaches on it and the hyper-parameters used during training. Finally, the subsection Construction Site Dataset describes dataset used and how the auxiliary task labels were generated.

3.1 Network Architecture

The architecture RefineNet, a state-of-the-art CNN for semantic segmentation [Lin *et al.*, 2017] is used as the base architecture for our investigations. The RefineNet framework uses multi-path refinement blocks over of the ResNet-101 architecture to fuse low-level features with high-level features. The fusion of these different levels of features enables improved localisation and classification of objects in images. For more details of RefineNet, please refer to [Lin *et al.*, 2017] and Github repository at *https://github.com/guosheng/refinenet*. Next, RefineNet is fine-tuned to only perform pixel-wise trip hazard detection, as a baseline approach (see section 3.3 for more details on network training). RefineNet is then trained to perform pixel-wise trip detection with each of the three different auxiliary tasks. These three tasks are described in the next section, Auxiliary Tasks.

3.2 Auxiliary Tasks

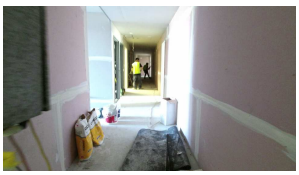
The three different auxiliary tasks investigated can be categorised into one classification and two regression tasks. Pixel-wise ground plane estimation is posed as the classification task as it involves predicting discrete values (ground plane, trip hazard or neither). While the pixel-wise depth and height above ground tasks are posed as regression tasks as they require estimation of continuous values.

When training for ground plane estimation, an additional class output is added to RefineNet, allowing it to perform pixel-wise trip and ground plane estimations using one Softmax Loss Function. When evaluating the trip detection of this auxiliary task, only the trip and non-trip confidences were considered. To perform trip detection with the two regression tasks an additional branch with two RCU blocks (Figure 3) were added after the final RefineNet block in parallel to the classification branch (see Figure 2). A L2 Regression Loss Function

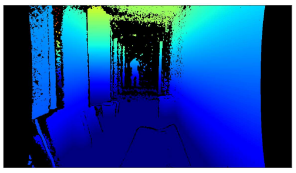
(a) 1st Floor (b) 2nd Floor

(c) 4th Floor (d) 7th Floor

Figure 4: Sample images of each from the four floors of the construction site dataset used. This dataset was released in earlier research [McMahon *et al.*, 2017]. Available at *http://tinyurl.com/Tripdataset*.



(a) RGB Image (b) Depth Image



(c) Height Image (d) Ground Plane Image

Figure 5: The four image types used, RGB, Depth, Height above ground and ground plane images. The height image was generated by plane fitting in the depth image. The ground plane images are a mask over points on the ground plane which are not trip hazards.

was used on this additional branch to train height above ground and the depth estimation tasks.

3.3 Training Procedure

This section describes the Data Prepossessing, Data Augmentation, Network Weight Initialisation, Hyper-Parameters and Cross-Validation and Testing approaches used to train the CNNs for our investigation.

Data Preprocessing

The depth and height images were converted into meters and then inverted ($d = 1/d$) before being used as labels. The set pixels with no depth or height values were set to be -1, and the loss at these points was ignored.

Data Augmentation

To augment the data all the images were randomly flipped horizontally and cropped during training. Images were also re-sized to 512 by 512 down from 960 by 540.

Weight Initialisation

The layers in the RefineNet CNN originally from ResNet-101 were initialised from an Image-Net model trained for image classification. While all the additional layers used in the RefineNet and RCU blocks (see Figure 3) were randomly initialised from a normal distribution with mean of 0 and standard deviation of 0.01.

Hyper-parameters

Scholastic Gradient Descent (SGD) with a batch size of 1 was used to train the CNNs. Learning rates of $1e^{-4}$ and $1e^{-5}$ were experimented with, but found to have negligible differences on the final performance. A momentum of 0.9 and a Weight Decay of $5e^{-4}$ was used. A Softmax Cross Entropy Loss Function was used for all classification tasks and a L2 (squared difference) Loss Function was used for the regression auxiliary tasks (see Section 3.2).

Cross-Validation and Testing

With the network training parameters chosen, the approaches are trained and then tested on the construction site dataset, described in Section 3.4. A four fold cross validation of the dataset is performed to more effectively leverage the limited amount of training data. For each fold, the approach is tested on one of the floors and trained on the other three floors of the construction site. All approaches are evaluated on each fold and the averaged performance is shown in Section 4.

3.4 Construction Site Dataset

Our approaches are trained and evaluated on the dataset from [McMahon *et al.*, 2017], recorded on a live construction site spanning $2000m^2$ with over 600 labelled trip hazards. The dataset is recorded over four floors, the first two floors are car parks, and the second two floors are apartment blocks (See Figure 4).

This construction site dataset consists of RGB and Depth images recorded from a Kinect 2, as well as HHA images. HHA images are three channel processed depth images [Gupta *et al.*, 2014]. HHA stands for Horizontal disparity, Height above ground, and Angle the pixel normal makes with the inferred gravity direction. During earlier work in [McMahon *et al.*, 2017] we found the assumption made by Gupta *et al.* that the ground plane is the lowest point in the image to be invalid for our dataset, due to the noise from the depth sensor. As a

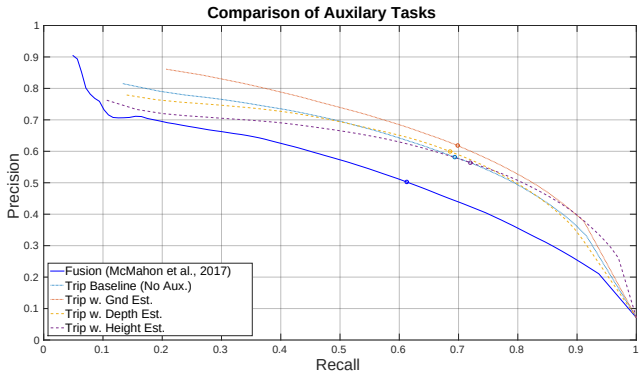


Figure 6: Precision-recall curve which compares the trip detection performance of the three different auxiliary tasks to our baseline approach and earlier work [McMahon *et al.*, 2017]. Circles represent the points shown in Table 1.

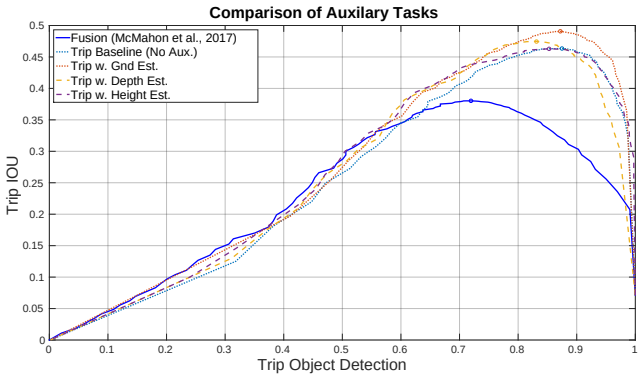


Figure 7: Shows the trade-off between Trip Object Detection and Trip IOU at different operating points. Circles represent the points shown in Table 1.

result of this, we decided to calculate the ground plane by processing the depth image ourselves with plane segmentation.

Ground Plane Label Generation

To generate the ground plane and height above ground plane auxiliary labels, a point cloud plane segmentation approach was used. This was done by first projecting the depth image to a 3D point cloud and running pass through filters to clean the point cloud. A RANSAC (Random Sample Consensus) approach was used over the point cloud to fit the largest horizontal plane. To reduce the search space and increase robustness we constrained the RANSAC search to regions +- 0.3 meters from the the height of the trip-pod (1.75m) used to mount the depth sensor. With the ground plane found, we calculated the distance between each point and the ground plane. To account for the noise from the Kinect 2, points found to be 0.1 meters above or below the plane were considered to be on the ground. We then re-projected the cloud back into the image space, but saved each points distance from the ground plane instead of it distance from the sensor. Sample height above ground and ground plane images can be seen in Figure 5.

4 Evaluation

4.1 Performance Metrics

To evaluate the trip detection rates, five performance metrics for semantic segmentation were used. The five metrics are Precision, Recall, F1-Score, Trip IOU and Trip Object Detection. Precision, Recall, Recall, F1-Score and Trip IOU are commonly used with semantic segmentation and are more formally defined in [McMahon *et al.*, 2017]. However, Trip Object Detection is a custom metric which measures the object-wise accuracy of the approaches, where the number of correctly predicted trip hazard objects is divided by the total number of labelled trip hazard objects. The Trip Object Detection metric is useful to determine if a given approach identifies most of the trip hazards in a scene, rather than how well the trip prediction pixels overlay with the positively labelled trip pixels.

The Root Mean Squared Error (RMSE) is used to evaluate the performance of the regression auxiliary tasks (depth and height estimation), as seen in Table 2. It is defined as $\sqrt{\frac{1}{n}\sum_i^n (d_i - \hat{d}_i)^2}$ where d is the ground truth depth, $\hat{d}$ is the estimated depth and n is the number of points in all images evaluated. A lower value is better.

4.2 Results and Discussion

The key results of this research can be seen in Figure 6 and 7 as well as Table 1 and include:

- Achieved state-of-the-art trip detection performance when estimating the ground plane as an auxiliary task.
- Established that the depth estimation task outperforms the height auxiliary task and the baseline approaches.
- Shows the height above ground auxiliary tasks, achieves higher recalls than baseline approach for lower precision values.
- Demonstrates all approaches introduced outperform an earlier approach which fuses colour and depth information as inputs to a CNN [McMahon *et al.*, 2017].

Figures 6 and 7, and Table 1 show the ground plane auxiliary task outperform all other approaches in majority of operating points. Figure 7 shows the ground plane estimation auxiliary task achieves the highest Trip IOU

(a) Colour Image (b) Ground Truth (c) Trip Pred Only (No Aux.) (d) Trip w. Ground Est.

Figure 8: The ground plane auxiliary task leverages knowledge of scene geometry to better infer trip hazards. This can be seen in the orange rectangle, where a region off the ground is falsely identified by the baseline approach (c) but is corrected with the addition of the ground plane estimation (d). Another example is highlighted in the blue rectangle, the addition of ground plane estimation (d) enables better identification of trip hazards, decreasing the number of false negative trip predictions.

(a) Colour Image (b) Ground Truth (c) Trip Pred Only (No Aux.) (d) Trip w. Ground Est.

Figure 9: Shows the addition of ground plane estimation enables our approach to better differentiate between trip hazards and ground, reducing the amount of false positive trip predictions.

metrics while maintaining a higher Trip Object Detection score than all other approaches. Figures 8 and 9 show instances of the ground plane estimation approach successfully leveraging information about the inferred scene geometry to improve trip detection performance. Figure 8 shows two examples of this, in the first (orange rectangle), a false trip prediction above the ground is corrected. In the second example (blue rectangle), a trip hazard is more comprehensively identified, to boost the true positive trip detection rate.

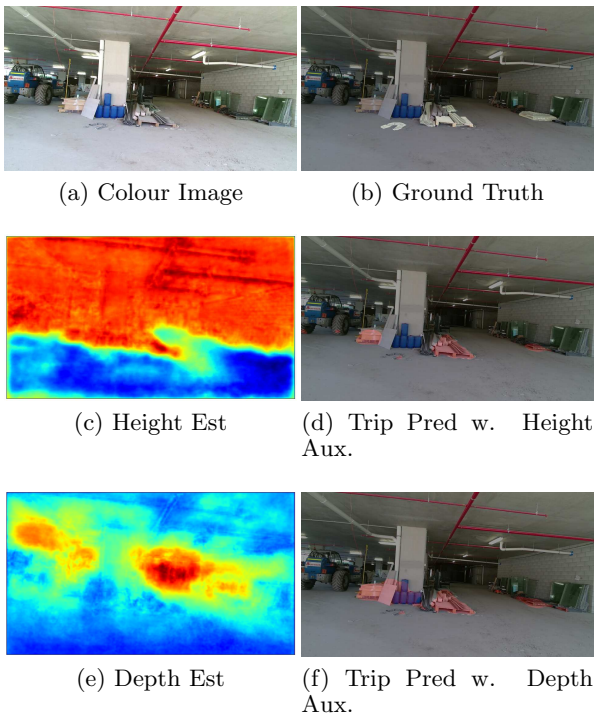
There are a number of explanations for why the ground plane auxiliary task outperforms the depth and height above ground auxiliary tasks. One explanation is the different kind of learning objectives being performed, ground plane estimation is posed as a classification objective while the other two are posed as regression objective. Regression tasks are typically much harder objectives to train for as the network has to learn to predict a much wider range of continuous values. While for our classification objective, the network only has to learn three discrete values at each pixel, trip hazard, ground plane or neither. Another explanation is the pretrained weights were trained for a classification tasks, and therefore already contains a set of features optimised to perform classification. Perhaps with more training data there would be more opportunity for the network to develop features better suited for *both* pixel-wise trip classification and depth or height estimation.

Table 1: Shows the performance of the trip hazard detectors. The operating points which give the highest F1-score are shown here.

Approach	Prec	Rec	F1	Trip IOU	Trip Obj.
Fusion (McMahon *et al.*, 2017)	50.2	61.3	54.8	38.0	72.0
Trip Baseline (No Aux.)	58.2	69.4	63.0	46.3	**87.5**
Trip w. Height Est.	56.4	**72.0**	62.6	46.3	85.3
Trip w. Depth Est.	59.9	68.6	63.7	47.5	83.1
Trip w. Gnd Est.	**61.8**	69.9	**65.5**	**49.1**	**87.2**

Of the other two auxiliary tasks, it can be seen the depth estimation auxiliary task outperforms height estimation and the baseline approach. This can be seen in Table 1 where the depth estimation auxiliary task achieves a higher Precision, F1 and Trip IOU scores. The same trend can be seen in Figures 6 and 7 with the depth auxiliary task outperforming height above ground estimation and the baseline approach in the majority of operating points. Further visualisations of the outputs from these two regression tasks can be seen in Figure 10.

The results also show all proposed approaches significantly outperforming an earlier work which fuses colour and depth information as inputs to a CNN [McMahon *et al.*, 2017]. All approaches outperform the fusion approach across all metrics and across all operating points.



(a) Colour Image (b) Ground Truth

(c) Height Est (d) Trip Pred w. Height Aux.

(e) Depth Est (f) Trip Pred w. Depth Aux.

Figure 10: Shows example predictions from the height above ground and depth auxiliary tasks.

5 Conclusion

The results presented in this research show improved trip hazard detection with the addition of auxiliary tasks. Through this investigation into three different auxiliary tasks, it was found that performing trip detection with ground plane estimation gives best performance. The ground plane estimation auxiliary achieved the highest F1 and Trip IOU scores (Table 1) as well as outperforming all other approaches in the majority of operating points (Figure 6 and 7).

With significant improvement over earlier work, this research makes it plausible to deploy an assistive robotic platform capable of helping human workers on construction sites. As in other fields where robots have been deployed, we believe that the addition of a consistent and reliable automation system to hazard detection will increase the effectiveness and safety of humans in performing their jobs.

Table 2: Shows the performance of the three auxiliary tasks. Root Mean Squared Error (RMSE) is used for the two regression tasks, where a lower value is better. To measure the performance of pixel-wise ground plane classification the IOU for the ground plane class is used, where a higher value is better.

Task	RMSE	Gnd IOU
Depth Est.	0.7815	-
Height Est.	0.3952	-
Gnd Plane Est.	-	49.4

References

[Chan *et al.*, 2015] Wesley P. Chan, Matthew K.X.J. Pan, Elizabeth A. Croft, and Masayuki Inaba. Characterization of handover orientations used by humans for efficient robot to human handovers. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1–6. IEEE, September 2015.

[Chao *et al.*, 2015] Yu-wei Chao, Zhan Wang, Rada Mihalcea, and Jia Deng. Mining Semantic Affordances of Visual Object Categories. *CVPR*, 2015.

[Chen *et al.*, 2015] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs. *International Conference on Learning Representations (ICLR)*, pages 1–14, 2015.

[Eigen and Fergus, 2015] David Eigen and Rob Fergus. Predicting Depth, Surface Normals and Semantic Labels with a Common Multi-Scale Convolutional Architecture. 2015.

[Fouhey *et al.*, 2015] David F Fouhey, Xiaolong Wang, and Abhinav Gupta. In Defense of the Direct Perception of Affordances. 2015.

[Gibson, 1978] James J Gibson. The ecological approach to the visual perception of pictures. *Leonardo*, 11(3):227–235, 1978.

[Goncalves *et al.*, 2014] Afonso Goncalves, Joao Abrantes, Giovanni Saponaro, Lorenzo Jamone, and Alexandre Bernardino. Learning intermediate object affordances: Towards the development of a tool concept. In *International Conference on Development and Learning and on Epigenetic Robotics*, pages 482–488. IEEE, October 2014.

[Gupta *et al.*, 2009] Abhinav Gupta, Aniruddha Kembhavi, and Larry S Davis. Observing human-object interactions: using spatial and functional compatibility for recognition. *IEEE transactions on pattern analysis and machine intelligence*, 31(10):1775–89, October 2009.

[Gupta *et al.*, 2014] Saurabh Gupta, Ross Girshick, Pablo Arbeláez, and Jitendra Malik. Learning Rich Features from RGB-D Images for Object Detection and Segmentation. *arXiv preprint arXiv:1407.5736*, pages 1–16, 2014.

[Hassan and Dharmaratne, 2015] Mahmudul Hassan and Anuja Dharmaratne. Attribute Based Affordance Detection from Human-Object Interaction Images. In Fay Huang and Akihiro Sugimoto, editors, *Image and Video Technology PSIVT 2015 Workshops*, volume 9555 of *Lecture Notes in Computer Science*, pages 220–232. Springer International Publishing, 2015.

[He and Girshick, 2017] Kaiming He and Ross Girshick. Mask R-CNN. 2017.

[He *et al.*, 2015] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. *Arxiv.Org*, 7(3):171–180, 2015.

[Hoffman *et al.*, 2016] Judy Hoffman, Trevor Darrell, and U C Berkeley. Learning with Side Information through Modality Hallucination. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 826–834, 2016.

[Kendall and Gal, 2017] Alex Kendall and Yarin Gal. What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? In *Arxiv*, March 2017.

[Kjellström *et al.*, 2011] Hedvig Kjellström, Javier Romero, and Danica Kragić. Visual object-action recognition: Inferring object affordances from human demonstration. *Computer Vision and Image Understanding (CVPR)*, 115(1):81–90, January 2011.

[Koppula and Saxena, 2015] Hema Koppula and Ashutosh Saxena. Anticipating Human Activities using Object Affordances for Reactive Robotic Response. *Transactions on Pattern Analysis and Machine Intelligence*, 8828(c):1–1, 2015.

[Li *et al.*, 2015] Bo Li, Chunhua Shen, Yuchao Dai, Anton Van Den Hengel, and Mingyi He. Depth and surface normal estimation from monocular images using regression on deep features and hierarchical CRFs. *Computer Vision and Pattern Recognition (CVPR)*, 2015.

[Liang *et al.*, 2015] Xiaodan Liang, Yunchao Wei, Xiaohui Shen, Jianchao Yang, Liang Lin, and Shuicheng Yan Senior. Proposal-free Network for Instance-level Object Segmentation. *TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, pages 1–14, 2015.

[Lin *et al.*, 2017] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid. RefineNet: Multi-Path Refinement Networks for High-Resolution Semantic Segmentation. In *Computer Vision and Pattern Recognition (CVPR)*, November 2017.

[Long *et al.*, 2015] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation. *Computer Vision and Pattern Recognition (CVPR)*, November 2015.

[Mar *et al.*, 2015] Tanis Mar, Vadim Tikhanoff, Giorgio Metta, and Lorenzo Natale. Self-supervised learning of grasp dependent tool affordances on the iCub Humanoid robot. In *International Conference on Robotics and Automation (ICRA)*, pages 3200–3206. IEEE, May 2015.

[McMahon *et al.*, 2015] Sean McMahon, Niko Sunderhauf, Michael Milford, and Ben Upcroft. TripNet: Detecting Trip Hazards on Construction Sites. *Australasian Conference on Robotics and Automation (ACRA)*, November 2015.

[McMahon *et al.*, 2017] Sean McMahon, Niko Sünderhauf, Ben Upcroft, and Michael Milford. Multi-Modal Trip Hazard Affordance Detection On Construction Sites. *IEEE Robotics and Automation Letters (RA-L)*, 3(1):1–8, 2017.

[Mirowski *et al.*, 2017] Piotr Mirowski, Razvan Pascanu, Fabio Viola, Hubert Soyer, Andrew J Ballard, Andrea Banino, Misha Denil, Ross Goroshin, Laurent Sifre, Koray Kavukcuoglu, and Deepmind London. Learning to Navigate in Complex Environments. *Under Review at International Conference on Learning Representations (ICLR)*, 2017.

[Moldovan and De Raedt, 2014] Bogdan Moldovan and Luc De Raedt. Occluded object search by relational affordances. In *International Conference on Robotics and Automation (ICRA)*, pages 169–174. IEEE, May 2014.

[Moldovan *et al.*, 2012] Bogdan Moldovan, Plinio Moreno, Martijn van Otterlo, Jose Santos-Victor, and Luc De Raedt. Learning relational affordance models for robots in multi-object manipulation tasks. In *2012 IEEE International Conference on Robotics and Automation*, pages 4373–4378. IEEE, May 2012.

[Oliveira *et al.*, 2016] Gabriel L. Oliveira, Wolfram Burgard, and Thomas Brox. Efficient deep models for monocular road segmentation. In *International Conference on Intelligent Robots and Systems (IROS)*, pages 4885–4891. IEEE, October 2016.

[Ren *et al.*, 2015] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *Advances in Neural Information Processing Systems*, pages 91–99, 2015.

[Tikhanoff *et al.*, 2013] V Tikhanoff, U Pattacini, Ieee Member, L Natale, Ieee Member, G Metta, and Ieee Senior Member. Exploring affordances and tool use on the iCub. *International Conference on Humanoid Robots (Humanoids)*, 2013.

[US Department of Labor, 2016] US Department of Labor. Occupational Injuries and Illnesses (Annual)

News Release. Technical report, US Department of Labor, 2016.

[Yao *et al.*, 2013a] Bangpeng Yao, Yao Bangpeng, Ma Jiayuan, and Fei-Fei Li. Discovering Object Functionality. In *2013 {IEEE} International Conference on Computer Vision*, 2013.

[Yao *et al.*, 2013b] Bangpeng Yao, Jiayuan Ma, and Li Fei-Fei. Discovering Object Functionality. In *2013 IEEE International Conference on Computer Vision (ICCV)*, pages 2512–2519. IEEE, December 2013.

[Yi *et al.*, 2015] Chang'an Yi, Huaqing Min, and Ronghua Luo. Affordance matching from the shared information in multi-robot. In *2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 66–71. IEEE, December 2015.