Training Deep Learning Tree Detection Algorithms using Synthetic Forest Images

Shahin Bano[1], J.P Singh[2]

[1] Department of Computer Science and Engineering, CEC Bilaspur
[2] Assistant Professor Department of Computer Science and Engineering, CEC Bilaspur

---***---

Abstract—Vision-based division in forested situations may be a key usefulness for independent ranger service operations such as tree felling and sending. Profound learning calculations illustrate promising comes about to perform visual assignments such as protest discovery. In any case, the directed learning handle of these calculations requires explanations from a expansive differences of pictures. In this work, we propose to utilize recreated woodland situations to naturally create 43k reasonable engineered pictures with pixel-level comments, and utilize it to prepare profound learning calculations for tree discovery. This permits us to address the taking after questions: i) what kind of execution ought to we anticipate from profound learning in unforgiving synthetic forest situations, ii) which explanations are the foremost critical for preparing, and iii) what methodology ought to be utilized between RGB and profundity. We moreover report the promising exchange learning capability of highlights learned on our manufactured dataset by specifically foreseeing bounding box, division covers and keypoints on genuine pictures. Code available on GitHub (https://github.com/norlab-ulaval/PercepTreeV1).

I. INTRODUCTION

Deep learning picked up much consideration within the field of ranger service because it can actualize information into machines to handle issues such as tree discovery or tree health/species classification [1]. In any case, profound learning could be a information centric approach that needs a sufficient amount of commented on pictures to memorize unmistakable protest highlights. Making an picture dataset may be a awkward handle requiring a awesome bargain of time and human assets, particularly for pixel-level comments. In like manner, few datasets particular to ranger service exist, and this limits profound learning applications, as well as errand robotization requiring high-level cognition.

In arrange to maintain a strategic distance from hand-annotation and incorporate as numerous reasonable conditions as conceivable in pictures, we propose to fill the information hole by making a expansive dataset of manufactured pictures containing over 43k pictures, which we title the SYNTHTREE43K dataset. Based on this dataset, we prepare Veil R-CNN [2], the foremost commonly-used show for occurrence division [1], and degree its exhibitions for tree location and division. Since our test system permits for speedy explanation, we moreover explore with keypoint location to supply data around tree breadth, slant and felling cut area.

Indeed in spite of the fact that the discovery exhibitions gotten on SYN-THTREE43K will not straightforwardly exchange to genuine world pictures since of the reality hole, a result investigation can direct us towards building an ideal genuine dataset. Eminently, manufactured datasets can be utilized to assess preparatory models [3], and some of the time they can make strides discovery execution when combined with real-world datasets [3], [4]. In that sense, we shed light on which explanations are the foremost impactful on learning, and on the off chance that including the profundity methodology within the dataset is relevant. Finally, we illustrate the reality hole by subjectively testing the show on genuine pictures, appearing exchange learning potential.

II. RELATED WORK

Deep learning for tree location in ranger service has illustrated victory on generally little genuine picture datasets. For occasion, [5] actualize a U-Net engineering to perform tree specie classification, discovery, division and stock volume estimation on trees. When prepared on their (private) dataset of 3k pictures, they accomplish 97.25% exactness and 95.68% review rates. Essentially, [6] employments a blend of unmistakable and warm pictures to form a dataset of 2895 pictures extricated from video groupings, and exclusively incorporate bounding box explanations. They prepared five distinctive one-shot locators on their dataset and accomplished 89.84% exactness, and 89.37% F1-score. We accept these previously mentioned strategies seem advantage from preparing on manufactured pictures. The Virtual KITTI dataset [3] is one of the primary to investigate this approach to prepare and assess models for independent driving applications. By reproducing real-world recordings with a diversion motor, they create engineered information comparable to genuine information. The models prepared on their virtual dataset appear that the hole between genuine and virtual information is little, and it can substitute for information holes in multi-object following. In the mean time, [7] explore protest discovery employing a engineered dataset for independent driving, and they report that preparing models on practically rendered pictures might create great segmentations by themselves on genuine datasets whereas significantly expanding exactness when combined with genuine information. Quantitatively, they made strides perclass exactness by more than 10 focuses (and in a few cases, as distant as 18.3 focuses).

In spite of the fact that engineered datasets cannot totally supplant genuine world information, different works illustrate that it could be a costeffective elective that provides great transferability [3], [4], [7]. In this manner, creating engineered woodland datasets will possibly progress the current state of tree recognition strategies in ranger service.

III. METHODOLOGY

In this area, we detail how SYNTHTREE43K was made. At that point, we portray the profound learning design and spines, as well as the preparing points of interest.

A. Simulator and Dataset

SYNTHTREE43K is produced by utilizing the Solidarity amusement motor to render reasonable virtual woodlands. This virtual world generator can be designed through Gaia to procedurally terra-form the scene, surface the territory and produce objects. In this reenactment, the timberland thickness is controlled through different produce rules such as height, landscape incline and the number of neighboring objects in a given region.

The woodland is populated with practical tree models from Nature Fabricate . In arrange to expand visual changeability, surface on the six tree models is adjusted to form 17 modern, particular tree models. Other protest models from Nature Fabricate are moreover included in scenes such as grass, stumps, scours and branches.

For extra authenticity and assortment, meteorological conditions are too reenacted in this virtual world. We recreate snow utilizing snow surface, and damp impact utilizing decals. Molecule frameworks are utilized to reproduce snowflakes, raindrops or haze impacts. To mimic distinctive minutes of the day, we alter brightening to morning, sunshine, evening and sunset. The protest shadows adjust to light cast on the scene.

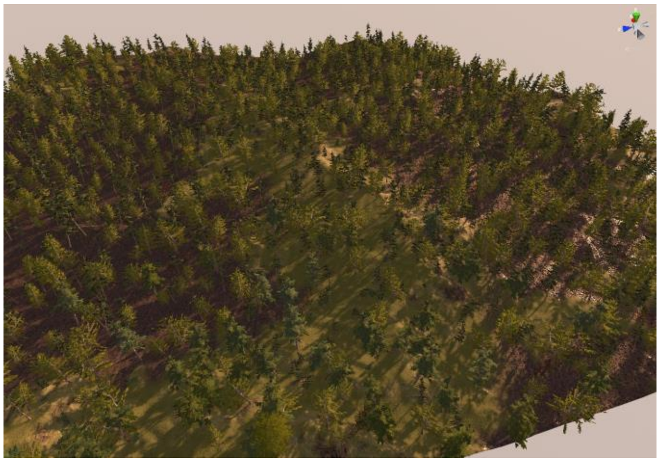

Fig. 3: A general view of a simulated forest environment. In this scene, three terrain textures are used to simulate moss, roots and mud conditions. The tree models are fir and beech, accompanied by scrubs, branches, grass and stomps under a morning light effect.

From each produced scene, we include between 200-1000 pictures to the dataset. Each picture incorporates bounding box, division veil and keypoint comments. The pipeline clarifies trees inside a 10m sweep from the camera, which compares to the reach of a gatherer [8]. Five keypoints are relegated per trees to capture the fundamental data that an independent tree felling framework would require: the felling cut area, breadth and slant.

Collectively, this pipeline can create an boundless sum of manufactured pictures with an explanation speed of roughly 20 frames/minute, for we consider RGB and profundity pictures as one outline. Generally, SYNTHTREE43K contains over 43k RGB and profundity pictures, and over 162k clarified trees.

B. Models

The Cover R-CNN engineering is composed of i) a convolutional highlight extraction spine, ii) a Locale Proposition Arrange (RPN), and iii) expectation heads. The initial Cover R-CNN arrange is somewhat adjusted for our tree discovery issue, in that, an discretionary keypoint department is included to the forecast head. In this manner, it can be utilized for classification, bounding box relapse, division, and keypoint forecast.

Expectations are made via a two-stage prepare. Within the to begin with arrange, the RPN proposes locales of interface (RoI) from the highlight maps of the spine. These compare to a locale that possibly contains a tree. The era of a RoI takes after along the default nine box grapples, comparing to three area-scales (8, 16 and 32) and three perspective proportions

(0.5, 1.0, and 2.0). In our try, we utilize three distinctive spine structures: ResNet-50, ResNet-101 [9], and ResNeXt-101 [10]. We utilize ResNet spine for include extraction because it gives amazing picks up in both exactness and speed, which we utilize as a baseline for our comes about. When comparing the 50-layer to its 101-layer partner, there's a plausibility that they perform so also when the dataset is little [2], however we anticipate that utilizing SYNTHTREE43K will be sufficient to illustrate that the 101-layer can outflank the 50-layer. In respects to ResNeXt, it presents a cardinality hyper-parameter, which is the number of free ways, giving a way to alter the demonstrate capacity without going more profound or more extensive. More points of interest approximately spine parameters are given in Table I

TABLE I: Backbone parameters. The number of learnable parameters (#Params), computational complexity (GFLOPs) and frames per second (FPS) at inference time on 800×800 images.

Backbone	#Params	GFLOPs	FPS
ResNet-50-FPN	25.6M	3.86	18
ResNet-101-FPN	44.7M	7.58	15
ResNeXt-101-FPN	44M	7.99	10

In this way, RoIAlign [2] employments bilinear insertion to outline the highlight maps of the spine into a 7Ã—7 input highlight outline inside each RoI zone. Highlights from each RoI at that point go through the arrange head to at the same time foresee the course, box balanced, double division veil, and an discretionary twofold veil for each keypoint.

C. Training Details

We utilize Detectron2 [11] executions of Veil RCNN. It has been appeared that pre-training makes a difference regularize models [12], and encourage exchange learning to a target space [13]. Hence, the Veil R-CNN models utilized in our tests are pre-trained on the COCO Individual Keypoint dataset [14], which could be a large-scale dataset containing more than 200k pictures and 250k individual occasions labeled with 17 keypoints per instance. Before preparing or fine-tuning the models, the primary two convolutional layers of the spine are solidified. The equipment for show preparing and testing is an NVIDIA RTX-3090-24GB GPU and an Intel Center i910900KF CPU.

To prepare the demonstrate, SYNTHTREE43K is part into three subsets: 40k within the prepare set, 1k for the validation set and 2k within the test set. The show learns from the prepare set by utilizing an stochastic slope plunge (SGD) optimizer with a energy of 0.9, and a weight rot of 0.0005. Amid preparing, we progress show generalization, and diminish dataset overfitting by utilizing information expansion procedures such as picture resizing, flat flipping, sheering, immersion, revolution, and trimming. No information enlargements are utilized at approval and test time. Show overfitting is checked by means of the approval set, which is additionally utilized for early halting.

Profundity pictures are gray scales of 8-bit 1-channel. At prepare time, they are changed over to 8-bit 3-channel tofit the RGB arrange from pre-trained models. In our case,asingle picture channel might be conceivable, but it wouldrequire haphazardly started models, which takes an colossal sum of pretraining time for spines such as ResNet and ResNeXt. Hyperparameter optimization is conducted for ResNet-50-FPN as it were, and these hyperparameters are utilized for each demonstrate. We utilize early ceasing based on the most noteworthy approval set Normal Accuracy (AP) to decide when to halt preparing.

IV. EXPERIMENTAL RESULTS

We base our execution assessment on the standard COCO measurements for each assignment, APbb and APmask, we prepare Cover R-CNN on our engineered woodland pictures and compare location execution between the three spines and RGB/Depth methodology. We moreover conduct an investigation of keypoint expectation by measuring the pixel mistake of each anticipated keypoint. In conclusion, we test discovery on genuine pictures, subjectively illustrating the reality crevice between manufactured and genuine pictures.

A. Tree Detection and Segmentation

Six models, comparing to all combinations between the three distinctive spines and two modalities, are prepared and tried on SYNTHTREE43K. From Table II, we watch that all models prepared on the profundity methodology beat models prepared on the RGB methodology. In truth, the discovery assignment based on the profundity methodology moves forward APbb by 9.49% on normal, indeed in spite of the fact that all of the models were pre-trained on COCO Individual â⍰” an RGB dataset. This proposes two things: 1) that the profundity methodology makes a difference systems dismiss trees found assist than our 10m explanation limit, and 2) profundity pictures are conceivably less demanding to decipher. In respect to the division assignment, exceptionally small pick up is gotten by utilizing profundity pictures rather than RGB, and within the case of ResNeXt-101 it diminishes. Shockingly, the ResNeXt design has more inconvenience exchanging to profundity pictures compared to the ResNet engineering, which make ResNet-101 the leading spine for profundity.

On RGB pictures, we accomplish the most excellent location comes about utilizing ResNeXt-101. This can be comparative to past investigate [10], [15], and it may be a result of the cardinality utilized in ResNeXt because it is more successful than going more profound or more extensive when the show capacity is expanded. The forecasts on manufactured pictures can be watched in Figure 1.

TABLE II: Results for models trained and tested on SYNTHTREE43K. All models achieved better performances using the depth modality.

Backbone	Modality	AP*bb*	AP*mask*	AP50^{bb}	AP50*mask*
R-50	RGB	55.20	31.13	87.74	69.36
	Depth	66.70	31.52	89.67	70.66
R-101	RGB	56.79	31.72	88.51	70.53
	Depth	68.20	31.98	89.89	71.65
X-101	RGB	58.34	31.77	88.91	71.07
	Depth	63.90	28.86	87.41	68.19

Table III appears that including the veil department reliably makes strides APbb and APkp. Comparatively, including the keypoint department diminishes APbb and APmask. These discoveries adjust with [2], as they found that the keypoint department benefits from multitask preparing, but it does not offer assistance the other assignments in return. Superior bounding box and keypoint location can happen by learning the highlights particular to division. In reality, a wealthier and point by point understanding of picture substance requires pixel-level division, which can play an imperative part in absolutely delimiting the boundaries of person trees [16].

TABLE III: Impact of multi-task learning on bounding box, segmentation mask, and keypoints. Results are from ResNeXt-101 on real RGB images.

Tasks	AP*bb*			AP*mask*AP*kp*
	59.25	**32.65**	-	
				31.77 80.19
	57.71	-	80.13	

B. Keypoint Detection

A keypoint discovery investigation based on mistake in pixels permits for important and direct elucidations of tree-felling errands and their mistake dissemination. In this manner, we compute the pixel mistake between the ground truth and the anticipated keypoint, and report the comes about in Figure 5.

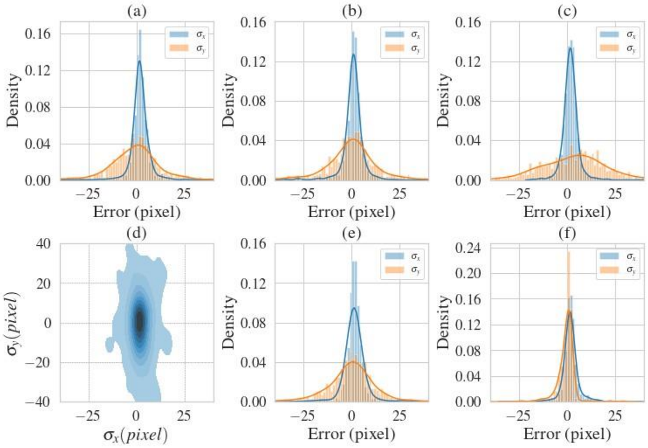

Fig. 5: Keypoint error distributions (in pixels) for our best detection model, ResNet-101 on depth images. Blue is the horizontal error and orange is the vertical error distribution for the (a) felling cut keypoint, (b) right and (e) left diameter keypoint; (c) middle keypoint and (f) top keypoint. Density map of the felling cut keypoint is shown in (d).

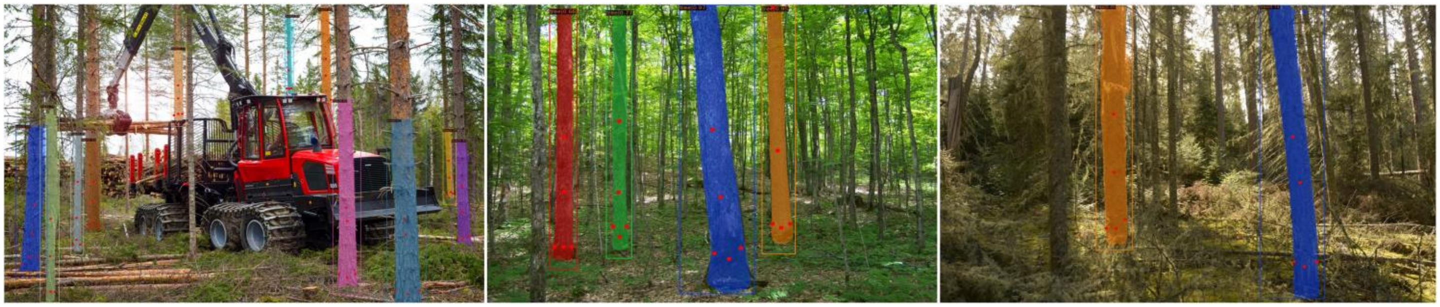

Fig. 4: Predictions on real images from ResNeXt-101 trained only on synthetic images. We observe that even with the reality gap, the model can still detect trees with high precision, but suffers from low recall rates.

Our tree location demonstrate accomplishes a cruel blunder of 5.2pixels when evaluating tree breadths. This precision is promising for robotization applications, depending on the separate between the tree and the camera. Since the change from pixel blunder to metric mistake depends on the profundity, the blunder increments when the tree is encourage absent, and in turn diminishes precision. We watch a critical contrast between even and vertical mistake, where Ï⍰y is approximately three times the esteem of Ï⍰x, for the felling cut, right and cleared out distance across keypoint, and center keypoint. We anticipated a bigger Ï⍰y than Ï⍰x, since the level keypoint position is either found on the side or center of the trees. In comparison, the vertical position of each keypoint is subjectively more troublesome to assess due to the failure of extricating exact vertical data. Consequently, the trouble in assessing the Ï⍰y blunder significantly impacts the assessed felling cut position. In the event that it is assessed as well tall on the stem, felling the tree will take off behind tall stumps that are against current gathering hones [17]. Keypoint expectations with tall vertical values regularly happen when a thick understorey limits the line of locate to the tree base. This causes flawed forecasts to position over the understorey, which comes about in an improper felling cut stature. In hone, a straightforward arrangement to this issue is to put the felling head on the anticipated point, and roll it down to the base [17].

C. Prediction on Real Images

We test the transferability of our demonstrate on genuine pictures. Due to the need of genuine picture datasets for tree discovery and division, the models are not fine-tuned on genuine pictures. Subjective comes about can be watched in Figure 4. Outwardly, we see that not as it were bounding boxes are well anticipated, but division veils beside keypoint forecasts are moreover exchanged effectively. The demonstrate appears to be more exact than precise, which demonstrates that it is incapable to distinguish trees that are as well diverse from the ones prepared on within the engineered dataset. Including distinctive tree models to our reenactment seem offer assistance generalize to the genuine world. Virtual pre-training may be a promising hone given the current information crevice in ranger service compared to other spaces, like independent driving or mechanical robotization.

V. CONCLUSION AND FUTURE WORKS

In brief, we investigated the utilize of engineered pictures to prepare profound learning calculations for tree discovery. We offer quantitative exploratory prove recommending that the division errand is vital and makes a difference to move forward both bounding box and keypoint expectations. Subsequently, the creation of a genuine picture dataset in ranger service ought to incorporate these comments. We too appear that the profundity methodology altogether outperforms the RGB methodology within the engineered world. At last, we subjectively illustrate that coordinate exchange to genuine world pictures endure from moo exactness, whereas the accuracy is generally great. Models freely accessible. In future works, we plan to evaluate tree detection performances on a real images dataset and assess its possible use in forestry related operations. In future works, we arrange to assess tree location exhibitions on a genuine pictures dataset and evaluate its conceivable utilize in ranger service related operations

REFERENCES

[1] Y. Diez, S. Kentsch, M. Fukuda, M. L. L. Caceres, K. Moritake, and M. Cabezas, “Deep learning in forestry using uav-acquired rgb data: A practical review,” *Remote Sensing*, vol. 13, no. 14, p. 2837, 2021.

[2] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask R-CNN,” in *IEEE International Conference on Computer Vision*, 2017, pp. 2961–2969.

[3] A. Gaidon, Q. Wang, Y. Cabon, and E. Vig, “Virtual worlds as proxy for multi-object tracking analysis,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4340–4349.

[4] C. R. de Souza12, A. Gaidon, Y. Cabon, and A. M. López, “Procedural generation of videos to train deep action recognition networks,” 2017.

[5] J. Liu, X. Wang, and T. Wang, “Classification of tree species and stock volume estimation in ground forest images using deep learning,” *Computers and Electronics in Agriculture*, vol. 166, p. 105012, 2019.

[6] D. Q. da Silva, F. N. Dos Santos, A. J. Sousa, and V. Filipe, “Visible and thermal image-based trunk detection with deep learning for forestry mobile robotics,” *Journal of Imaging*, vol. 7, no. 9, p. 176, 2021.

[7] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, “The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3234–3243.

[8] O. Lindroos, O. Ringdahl, P. La Hera, P. Hohnloser, and T. H. Hellström, “Estimating the position of the harvester head–a key step towards the precision

forestry of the future?” *Croatian Journal of Forest Engineering: Journal for Theory and Application of Forestry Engineering*, vol. 36, no. 2, pp. 147–164, 2015.

[9] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[10] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, “Aggregated residual transformations for deep neural networks,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1492– 1500.

[11] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, *Detectron2*, https://github.com/ facebookresearch/detectron2, 2019.

[12] D. Erhan, A. Courville, Y. Bengio, and P. Vincent, “Why does unsupervised pre-training help deep learning?” In *International conference on artificial intelligence and statistics*, JMLR Workshop, 2010, pp. 201– 208.

[13] D. Mahajan, R. Girshick, V. Ramanathan, K. He, M. Paluri, Y. Li, A. Bharambe, and L. Van Der Maaten, “Exploring the limits of weakly supervised pretraining,” in *European Conference on Computer Vision (ECCV)*, 2018, pp. 181–196.

[14] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, “Microsoft COCO: Common objects in context,” in *IEEE European Conference on Computer Vision (ECCV)*, Springer, 2014, pp. 740–755.

[15] M. Wu, H. Yue, J. Wang, Y. Huang, M. Liu, Y. Jiang, C. Ke, and C. Zeng, “Object detection based on rgc mask r-cnn,” *IET Image Processing*, vol. 14, no. 8, pp. 1502–1508, 2020.

[16] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, “Deep learning for generic object detection: A survey,” *International Journal of Computer Vision*, vol. 128, no. 2, pp. 261–318, 2020.

[17] D. Ireland and G. Kerr, “Ccf harvesting method development: Harvester head visibility,” 2008.