SAND2020-6001C

Synthetic Training Images for Real-World Object Detection

Zoe Gastelum, Timothy Shead, Michael Higgins

Sandia National Laboratories[1]

Albuquerque, NM, USA

ABSTRACT

Deep learning image classification and object detection models are widely used to identify and triage images for additional human review. Current open source licensed and commercial deep learning algorithms for understanding image content focus on commonly known items – plants, animals, buildings, sports equipment, etc. For many applications, these common set of classes may be suitable. However, multiple efforts within the nuclear nonproliferation community currently focus on applications of these algorithms for international safeguards to support the review of surveillance images, open source images, and images within an existing internal data repository. For safeguards applications, models must be fine-tuned to recognize the specific characteristics of relevant objects or classes. Research is ongoing to try to limit the numbers of images required to fine-tune these models, but typically several thousand examples of a single class are needed. Curating a dataset of this magnitude poses several challenges for international safeguards: 1) proliferation-relevant images may be rare due to their sensitivity or the limited availability of a technology; 2) creating relevant images through real-world staging is costly and introduces biases into the resulting model; and 3) expert-labeling is expensive, time consuming, and prone to error and dissent. To address these challenges, we are generating high-quality, three-dimensional computer graphic models which we use to render large numbers of images with high-variance lighting, background, perspective, and material properties that can be used to train deep learning models. In this paper, we will discuss our research goals, our experimental plan, and our results to-date using virtual, rendered images to train deep learning object detection models to recognize real-world images.

INTRODUCTION

The review of visual information is a significant aspect of international nuclear safeguards verification conducted by the International Atomic Energy Agency (IAEA). From the collection of open source multimedia data and satellite imagery to review of surveillance camera data and examination of seals for evidence of tamper, visual information abounds. Many safeguards activities that require visual inspection face an increase in data availability that out-paces the human ability for review. Consequently, the IAEA is considering computer vision algorithms to aid safeguards inspectors and analysts.

Computer vision collectively refers to a set of tasks that include determining what objects are in an image, where they are, how they relate to the background, and the meaning of the image as a whole. Computer vision has a long history in computing but has gained significant attention recently due to improvements in performance made possible by deep learning, large labeled datasets, and inexpensive computation. Two of the most widely known computer vision tasks are image classification and object detection, where

[1] Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525. SAND-XXXX

image classification predicts which of a finite set of labels best describe an entire image (Figure 1, left), while object detection labels multiple regions within an image, which are identified by bounding boxes or masks (Figure 1, right).

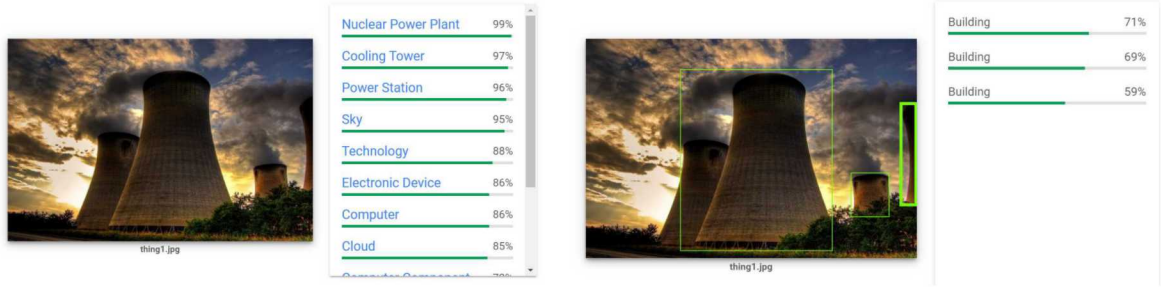

Figure 1: Image classification (left) versus object detection (right) using Google Cloud Vision API. Image credit: Jonathan Brennan, via Flicker. Image number 3635530311, 06/12/2009.

While deep learning approaches to computer vision have had impacts in many domains including medicine (Esteva et al. 2017; Antony et al. 2016), commerce (Kumar 2019), and self-driving cars (Bojarski et al. 2016), their use for international nuclear safeguards is still exploratory. An important bottleneck is that deep learning models can require thousands to millions of labeled observations for training, which requires considerable human expert time and is subject to human error and disagreement. Further, some safeguards applications lack sufficient training data due to the sensitivity of the objects to be detected, commercial proprietary concerns, or historical rarity of the objects in question. In addition, since deep learning computer vision models learn explicitly from existing examples, they may not recognize novel designs, configurations, proportions, or materials that are within the possibility of physics but not represented in the training data.

We hypothesize that synthetic images, generated from three-dimensional CAD models of an object of interest can be used during training to overcome these problems. Training images generated in this manner can be parameterized to produce an unlimited variety of designs, materials, and environments, and the ground truth in synthetic images can be explicit and unambiguous by-construction. Recent research on this topic includes the level of realism needed in the synthetic images (Tremblay et al. 2018), the integration of large real-world data sets with synthetic data (Ekbatani et al. 2017), and the use of synthetic images for classes which approximate common objects already in pre-trained models (Rahnemoonfar and Sheppard 2017). Our goal is to explore how synthetic images could be used to reduce – or ultimately eliminate – the requirement for real-world images during training.

SYNTHETIC IMAGE GENERATION

For our experiments we needed a nuclear-fuel-cycle-relevant subject that wasn't included in common image datasets such as ImageNet (Stanford Vision Lab 2016) or Common Objects in Context (Lin et al. 2014), but for which we could obtain a reasonable number of real-world images for model validation. Based on these criteria, we chose to model remote manipulator arms, specifically the master-side controllers which a human operator would use (rather than the robotic arms within hot cells) such as those used to separate plutonium isotopes.

We collected 220 images of manipulators from open sources (Figure 2), and 191 distractor images of related environments such as spent fuel pools, lab space, and industrial equipment. A subset of the images

were graciously provided by colleagues at Lawrence Livermore National Laboratory working on a nonproliferation and deep learning strategic initiative.[2]

Figure 2: Real-world image of remote manipulator arm. Image credit: Savannah River National Laboratory via Twitter: https://twitter.com/srnlab/status/983767392265306112

We identified eight visually distinct designs of manipulator arm and selected one of the more common designs to model: a long, telescoping tube with a single articulation point at the wall attachment and another at the operator handle. Based on images of this design, we constructed a parameterized three-dimensional model (Figure 3) using SideFX Houdini,[3] a popular procedural modeling and rendering tool used widely in the visual effects and gaming industries. By sampling from the parameters of the model, we could generate an unlimited number of poses and configurations. We used similar random sampling to generate a random camera angle for each image to be generated.

Figure 3: 3D manipulator model (left) with parameters (right).

Finally, to provide backgrounds combined with realistic lighting for the synthetic images, we sampled at random from a collection of 33 panoramic high dynamic range images depicting real world industrial environments such as a boiler room (Figure 4), machine shop, and pump house. Because the panoramic images depict full, 360-degree environments, they produce plausible backdrops for any configuration of randomly chosen camera angles.

[2] https://www.llnl.gov/news/researchers-developing-deep-learning-system-advance-nuclear-nonproliferation-analysis

[3] https://www.sidefx.com/products/houdini/

Figure 4: Sample panoramic high dynamic range image used as a synthetic backdrop.

While many real-world images of remote manipulator arms are associated with an iconic glow from the shielded hot cell (Figure 2), we opted not to generate hot cells in hopes that models trained using our data could support identification of manipulator arms outside of operational environments, such as shipment containers or prior to installation at a facility. We generated one thousand synthetic images of remote manipulator arms with the single-tube design, with varying manipulator pose, camera position, and choice of background and lighting (Figure 5). In addition, we generated one thousand background-only images, to act as distractors. All images were rendered at a resolution of 720 x 720 pixels using the Redshift[4] render engine.

Figure 5: Sample synthetic manipulator arm image.

BASELINE EXPERIMENTS: DATA AND MODEL VALIDATION

After preliminary experimentation with object detection models, we decided to focus on testing our synthetic data using image classification, with which we have greater familiarity. All of our experiments were conducted using a ResNet-50 image classification model (He et al. 2016), pretrained on ImageNet, and provided by the *torchvision.models* package included with PyTorch.[5] Since we were training models to detect a single class ("manipulators"), we replaced the final 1000-output fully connected layer in each model, substituting a 1-output fully connected layer with sigmoid activation function. This allowed us to use a value of zero as the ground truth for images without manipulators, and one for images with manipulators, treating the model's output predictions as confidence levels. For all of our experiments, we

[4] https://www.redshift3d.com

[5] https://pytorch.org/docs/stable/torchvision/models.html#classification

used a value of 0.5 as the output manipulator detection threshold for purposes of calculating performance metrics.

For these baseline experiments, we wanted to validate that we had enough real-world data and enough synthetic data for training models generally, that our model was architecturally sound, and identify a baseline for later results. To test this, we trained two sets of models: one trained and tested exclusively on real-world data, and the other trained and tested exclusively on synthetic data.

First, we fine-tuned the ResNet-50 model for 500 epochs using our 411 real-world images. We used 5x2 cross validation - holding back 20% of our training data for validation - resulting in an average test accuracy of 86% (Figure 6).

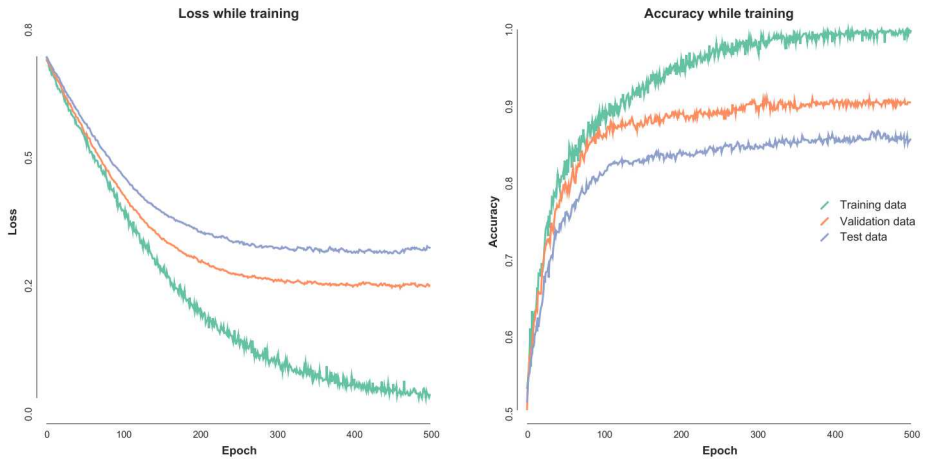

Figure 6: Average loss and accuracy training and testing using only real-world images with 5x2 cross validation.

We then performed a similar experiment exclusively using our 2000 synthetic images, resulting in cross validated test accuracy of 90% (Figure 7). The higher performance in this case was expected, due to the larger number of synthetic images. Note however that the validation loss in Figure 7 hits a low point around epoch 150 and increases thereafter, suggesting that the synthetic data was overfit, a pattern we will see again in later experiments.

Of course, neither of these experiments addressed our target of training using synthetic data to make predictions on real world images, but these results gave us confidence that our real-world and synthetic data sets were large enough for the fine-tuning process.

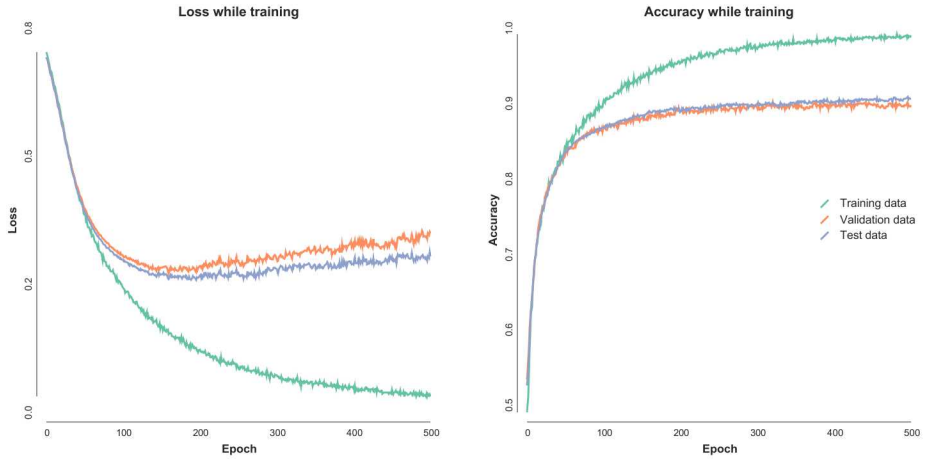

Figure 7: Average loss and accuracy training and testing using only synthetic images with 5x2 cross validation.

IMAGE BACKGROUND EXPERIMENTS

Once we had determined that we had sufficient data to fine-tune the models, the next step was to test the performance of models trained on synthetic data only and tested on real-world data. Ideally, the performance of the models trained using synthetic data would approach the performance of the models trained on real-world data.

Since the training and test sets were fixed in this case, 5x2 cross validation didn't make sense. Despite this, we used a random 20% of our synthetic data for validation, repeating each experiment ten times and averaging the results to avoid misleading results from random sampling. The results can be seen in Figure 8.

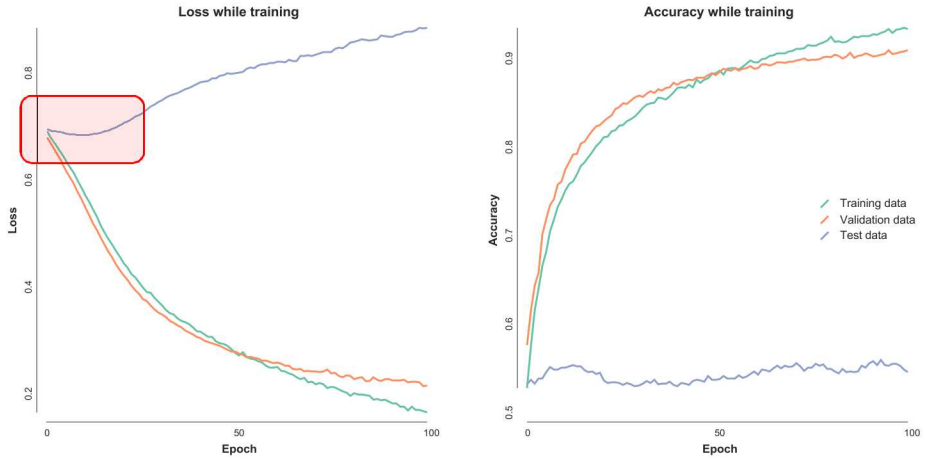

Figure 8: Average performance of synthetic training data with real test data, for ten models.

Here, our test accuracy is discouragingly low: around 55%, only slightly better than random. Interestingly, the test loss in Figure 8, highlighted in red, does decrease for the first ten epochs, meaning that training with synthetic data does provide some benefit early in the training process; however, the steadily increasing test loss in later epochs suggests the model has become overfit.

This led us to speculate that our synthetic data did not contain enough variety, compared to the real data. As a simple way to introduce additional variance, we re-rendered our 2000 synthetic images, but substituted randomly-selected backgrounds from our 191 real-world distractor images instead of our 33 panoramic HDR images. Because the real-world images weren't panoramic, they were unlikely to ever match our randomly chosen camera angles, but our hypothesis was that the benefit of the additional variance would outweigh the decrease in realism. The results of re-running the experiment with this hybrid synthetic-foregrounds-real-backgrounds dataset can be seen in Figure 9.

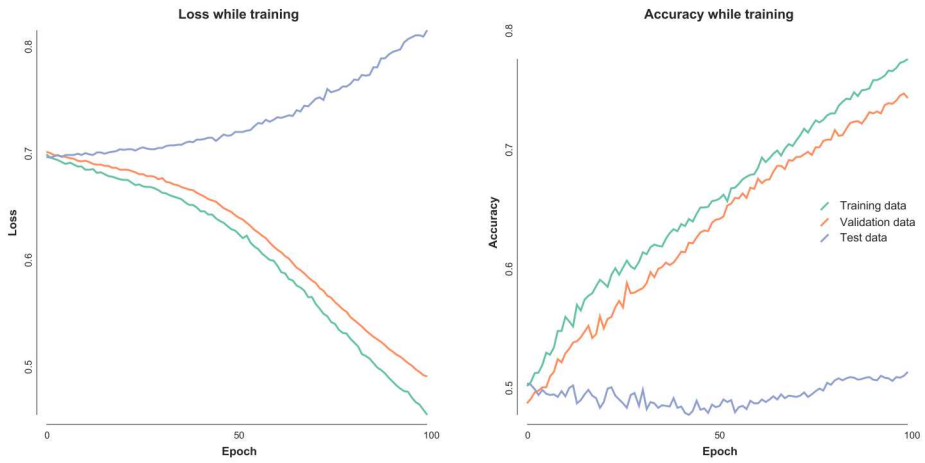

Figure 9: Average performance of synthetic-foreground + real-background training data, ten models.

Surprisingly, the addition of real background images to the synthetic manipulator images further reduces performance, to effectively random. Further, note that the test loss curve *never* decreases, suggesting that the hybrid dataset is of zero value when training to make predictions on real images.

REAL-SYNTHETIC RATIO EXPERIMENTS

Although our initial set of experiments was disappointing, we wanted to test the hypothesis that our synthetic data might be able to improve training outcomes when *supplementing* real world data rather than replacing it. To test this, we conducted a second round of experiments where we used a combination of real world and our (original, panoramic background) synthetic images to train classification models while steadily reducing the number of real-world images used for training. We compared the performance of these models against their real-world-only counterparts, to see whether the synthetic data improved performance. Note that the plots that follow show the minimum, average, and maximum values for loss and accuracy.

Figure 10 illustrates our first iteration, comparing the test loss and accuracy when training with 50% (205 images) of the available real-world data, versus 50% real-world images plus 100% (2000 images) of the synthetic data. As can be seen on the right side of the figure, the synthetic data provides an early boost in performance before converging to roughly the same performance (79% accuracy) as the real-world-data-only models.

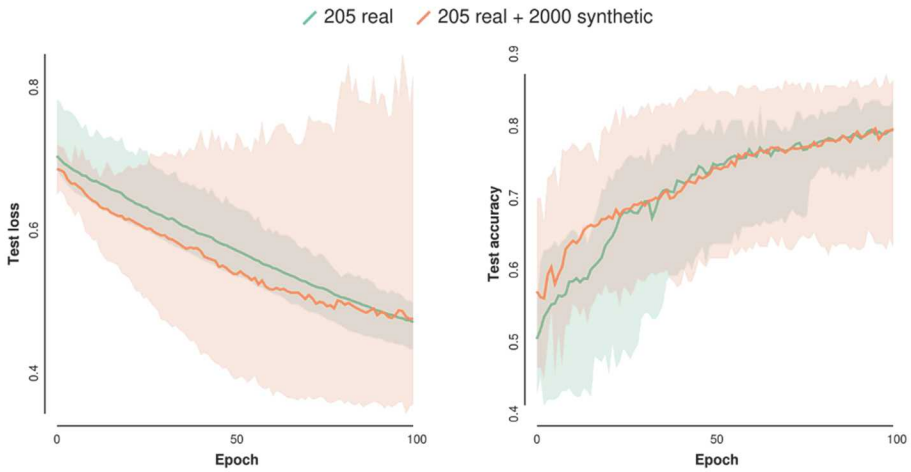

Figure 10: Real-only versus real+synthetic test performance averaged across ten models.

In Figure 11, the number of real-world images is decreased to 12.5% of the available data (51 images). Here, we see a more dramatic improvement in performance from the *real+synthetic* data. Interestingly, the variance in the *real+synthetic* results are much lower than the *real-only* results, which may be of value even if the results eventually converge in later epochs. The test loss for the *real+synthetic* data demonstrates the same early-dip behavior as in prior experiments.

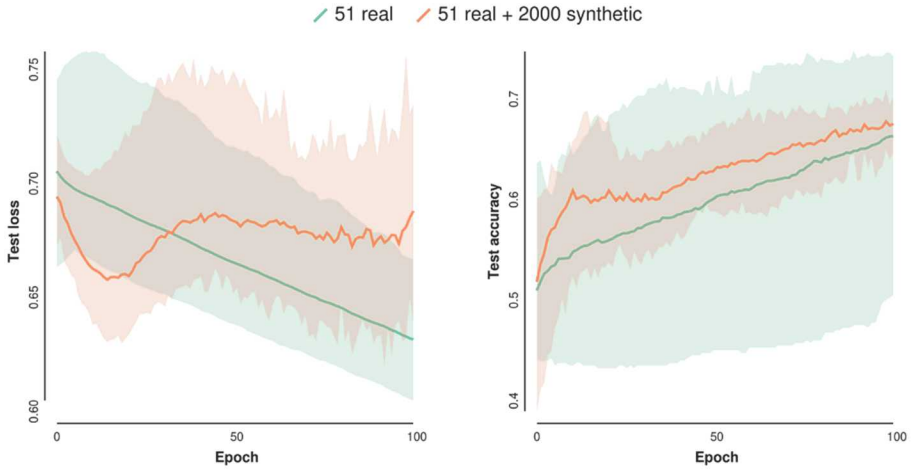

Figure 11: Real-only versus real+synthetic test performance averaged across ten models.

Finally, we decreased the number of real-world images is decreased to roughly 3% of the available data (12 images). Unsurprisingly, the performance of the models at this point is very similar to that of the earlier no-real-world training data experiment, although the variance of the *real+synthetic* accuracy is much better than real-world data alone as illustrated in Figure 12.

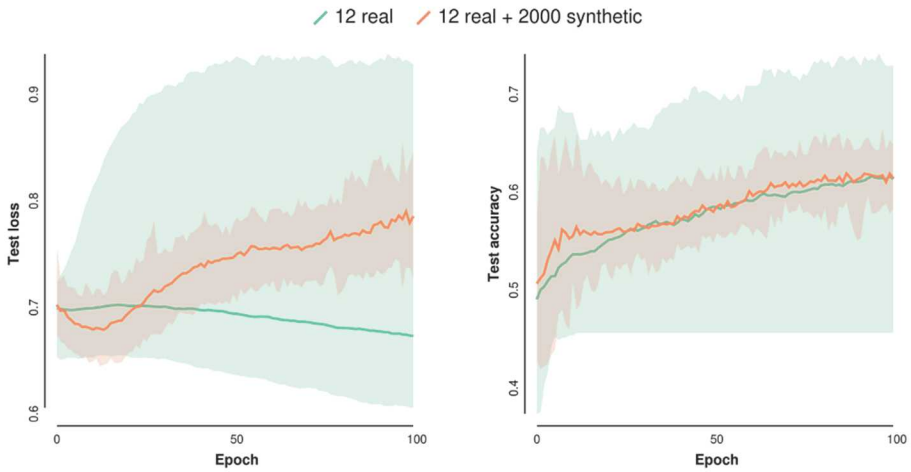

Figure 12: Real-only versus real+synthetic test performance averaged across ten models.

CONCLUSIONS

Although our proof-of-concept results are extremely preliminary, they do point to several interesting conclusions:

- From our results comparing panoramic to flat backgrounds, we speculate that the flat backgrounds led to lower performance because they rarely aligned with the randomly chosen synthetic camera angle. This suggests that realism may be more important than we originally imagined.
- Our synthetic data led to useful loss reduction for the real-world prediction task in early epochs, before diverging. This suggests that the distribution of our synthetic data needs to match the distribution of our real-world data more closely, again implying that more realism is better.
- Synthetic data in combination with real world data was useful when real world data was limited, and decreased variance in test accuracy even when there was no improvement in mean accuracy. This suggests that synthetic data could be useful for decreasing uncertainty even when sufficient real-world data is available (perhaps by training models on unlikely scenarios).

FUTURE WORK

Significant research remains to be done on the use of synthetic images to train object detection and classification algorithms for international safeguards. There are many unexplored avenues remaining in this work, such as the effect of different materials (paint, metal, plastic) on the synthetic models, including aging materials (such as rusty metal). We also plan to develop additional 3D models for the other manipulator designs appearing in our real-world data, and explore stratified training focusing on specific models of manipulator. The unusual results from our studies of different backgrounds suggest that they may play a larger role in performance than we originally imagined.

Assuming that we are able to improve the efficacy of our synthetic training data, additional experiments will be needed to determine whether synthetic imagery can generalize to other computer vision tasks such as object detection.

REFERENCES

Antony, Joseph, Kevin McGuinness, Noel E. O'Connor, and Kieran Moran. 2016. "Quantifying Radiographic Knee Osteoarthritis Severity Using Deep Convolutional Neural Networks." In *Proceedings - International Conference on Pattern Recognition*, 0:1195–1200. Institute of Electrical and Electronics

Engineers Inc. https://doi.org/10.1109/ICPR.2016.7899799.

Bojarski, Mariusz, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D. Jackel, et al. 2016. “End to End Learning for Self-Driving Cars,” April 2016. http://arxiv.org/abs/1604.07316.

Ekbatani, Hadi Keivan, Oriol Pujol, and Santi Segui. "Synthetic Data Generation for Deep Learning in Counting Pedestrians." In ICPRAM, pp. 318-323. 2017. DOI: 10.5220/0006119203180323

Esteva, Andre, Brett Kuprel, Roberto A. Novoa, Justin Ko, Susan M. Swetter, Helen M. Blau, and Kumar, Dilip. “Amazon Go: Origins and a Peek Under the Hood.” Oral presentation at Amazon re:MARS conference, June 13, 2019. Available at: https://www.youtube.com/watch?v=Lu4szyPjIGY Accessed 3/30/2020

Girshick, Ross, Radosavovic, Ilija, Gkioxari, Georgia, and Piotr Doll. “Detectron.” 2018. https://github.com/facebookresearch/Detectron

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770-778. 2016. Available at: https://arxiv.org/abs/1512.03385

Lin, Tsung-Yi, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. "Microsoft coco: Common objects in context." In *European conference on computer vision*, pp. 740-755. Springer, Cham, 2014. Available at: https://arxiv.org/abs/1405.0312

Rahnemoonfar, Maryam, and Clay Sheppard. "Deep count: fruit counting based on deep simulated learning." Sensors 17, no. 4 2017. https://doi.org/10.3390/s17040905

Redmon, Joseph, and Ali Farhadi. "Yolov3: An incremental improvement." *arXiv preprint arXiv:1804.02767* 2018. Available at: https://arxiv.org/abs/1804.02767

Sebastian Thrun. 2017. “Dermatologist-Level Classification of Skin Cancer with Deep Neural Networks.” *Nature* 542 (7639): 115–18. https://doi.org/10.1038/nature21056.

Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." *arXiv preprint arXiv:1409.1556* 2014. Available at: https://arxiv.org/pdf/1409.1556.pdf%20http://arxiv.org/abs/1409.1556.pdf

Stanford Vision Lab. “ImageNet.”2014. http://www.image-net.org/

Tremblay, J., Prakash, A., Acuna, D., Brophy, M., Jampani, V., Anil, C., et al. (2018, April 17). Training Deep Networks with Synthetic Data: Bridging the Reality Gap by Domain Randomization. arXiv.org.

Zhou, Bolei, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. "Learning deep features for discriminative localization." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2921-2929. 2016.