

Want computers to see better in the real world? Train them in virtual reality

April 11 2018

Scientists have developed a new way to improve how computers "see" and "understand" objects in the real world by training the computers' visual systems in a virtual environment.

The research team published their findings in *IEEE/CAA Journal of Automatica Sinica*, a joint publication of the IEEE and the Chinese Association of Automation.

For computers to learn and accurately recognize objects such as buildings, streets or humans, the machines must rely on processing huge amount of labeled data, in this case, images of objects with accurate annotations. A self-driving car, for instance, needs thousands of images of roads and cars to learn from. Datasets therefore play a crucial role in the training and testing of the computer [vision systems](#). Using manually labeled training datasets, a computer vision system compares its current situation to known situations and takes the best action, whatever that happens to be.

"However, collecting and annotating images from the real world is too demanding in terms of labor and money investments," writes first author Kunfeng Wang, an associate professor at China's State Key Laboratory for Management and Control for Complex Systems. Wang says the goal of their research is to overcome limitations of real-world image datasets, which are insufficient for training and testing computers vision systems.

To solve this issue, Wang and his colleagues created a [dataset](#) called

ParallelEye. ParallelEye was virtually generated by using commercially available computer software, specifically the video game engine Unity3D. Using a map of Zhongguancun, one of the busiest urban areas in Beijing, China, as their reference, they virtually recreated urban setting by adding buildings, cars and even weather conditions. Then they placed a virtual "camera" on a virtual car. The car drove around the virtual Zhongguancun and created datasets that are representative of the real world.

Through their "complete control" of the [virtual environment](#), Wang's team was able to create extremely specific usable data for their object detection system—a simulated autonomous vehicle. The results were impressive, producing a marked increase in performance on nearly every tested metric. By designing custom-made datasets, a greater variety of autonomous systems will be more practical to train.

While their greatest performance increases came from incorporating ParallelEye datasets with [real-world](#) datasets, Wang's team has demonstrated that their method is capable of easily creating diverse sets of images. "Using the ParallelEye vision framework, massive and diversified images can be synthesized flexibly, and this can help build more robust computer vision systems," says Wang. The research team's proposed approach can be applied to many visual computing scenarios, including visual surveillance, medical image processing, and biometrics.

Next, the team will create an even larger set of [virtual images](#), improve the realism of virtual images, and explore the utility of virtual images for other computer vision tasks. Wang says, "Our ultimate goal is to build a systematic theory of parallel vision able to train, test, understand and optimize [computer vision](#) models with virtual images and make the models work well in complex scenes."

More information: Yonglin Tian et al, Training and testing object

detectors with virtual images, *IEEE/CAA Journal of Automatica Sinica* (2018). [DOI: 10.1109/JAS.2017.7510841](https://doi.org/10.1109/JAS.2017.7510841)

Provided by Chinese Association of Automation

Citation: Want computers to see better in the real world? Train them in virtual reality (2018, April 11) retrieved 27 April 2024 from <https://phys.org/news/2018-04-real-world-virtual-reality.html>

<p>This document is subject to copyright. Apart from any fair dealing for the purpose of private study or research, no part may be reproduced without the written permission. The content is provided for information purposes only.</p>
--