Demo: A Distributed Virtual Vision Simulator

Wiktor Starzyk[1] Adam Domurad[2] Faisal Z. Qureshi[3]
Faculty of Science
University of Ontario Institute of Technology
Oshawa, ON L1H 7K4 Canada

***Abstract*—Realistic virtual worlds can serve as laboratories for carrying out camera networks research. This unorthodox "Virtual Vision" paradigm advocates developing visually and behaviorally realistic 3D environments to serve the needs of computer vision. Our work on high-level coordination and control in camera networks is a testament to the suitability of virtual vision paradigm for camera networks research. The prerequisite for carrying out virtual vision research is a virtual vision simulator capable of generating synthetic imagery from simulated real-life scenes. We present a distributed, customizable virtual vision simulator capable of simulating pedestrian traffic in a variety of 3D environments. Virtual cameras deployed in this synthetic environment generate synthetic imagery—boasting realistic lighting effects, shadows, etc.—using the state-of-the-art computer graphics techniques. The synthetic imagery is fed into a "real-world" vision pipeline that performs visual analysis—e.g., blob detection and tracking, facial detection, etc.—and returns the results of this analysis to our simulated cameras for subsequent higher level processing. It is important to bear in mind that our vision pipeline is designed to handle real world imagery without any modifications. Consequently, it closely mimics the performance of a vision pipeline that one might deploy on physical cameras. Our virtual vision simulator is realized as a collection of modules that communicate with each other over the network. Consequently, we can deploy our simulator over a network of computers, allowing us to simulate much larger networks and much more complex scenes then is otherwise possible.**

I. Virtual Vision Simulator

Virtual vision simulators are software laboratories for developing and evaluating camera networks. Qureshi and Terzopoulos [1], building upon the earlier work by Shao and Terzopoulos [2], developed a virtual vision simulator comprising a 3D reconstruction of the original Penn train station, populated with up to 1000 self-animating pedestrians. Their work on smart camera networks capable of carrying out observation tasks with minimal human supervision demonstrates the usefulness of this virtual vision simulator. A virtual vision simulator offers several advantages over traditional physical camera setups during ideation, prototyping, and evaluation phases, including:

- legal issues, such as privacy, surrounding access to camera networks installed in public spaces, disappear when using a virtual vision simulator;
- low cost—a virtual vision simulator runs on standard PCs and does not require any special hardward;
- quick prototyping—it is much easier and faster to reconfigure a virtual camera network than to reconfigure a physical camera network;
- complex vision and control algorithms that need to be studied in "real time" can be easily studied in a virtual vision simulator by slowing down the virtual clock of the simulated environment;
- faster design/evaluate iterations;
- ground truth is readily available; and
- full control over every aspect of the environment.

Fig. 1: A view of our virtual world showing pedestrians walking on an upper floor of an office building. Toronto (Canada) skyline is visible through floor to ceiling windows at the back. Our scripted pedestrians use motion-capture data to simulate realistic motion and cast dynamic shadows on the floor and the walls.

Here, we present a distributed virtual vision simulator. Our simulator is designed to address some of the shortcomings of the virtual vision simulator employed by Qureshi and Terzopoulos in their work on smart camera networks. Specifically, their virtual vision simulator is tied to a single PC; whereas, our simulator can be deployed over a network of computers. The ability to spread the computational load over multiple computers assumes urgency as we begin to simulate richer, more complex synthetic worlds and larger, more sophisticated virtual camera networks. Unlike the virtual vision system used by Qureshi and Terzopoulos, our simulator uses state-of-the-art graphics technology to support advanced rendering effects—such as lighting, shadows, transparency—which adds to the visual realism of the scene (Fig. 1). Lastly, we have a developed a vision pipeline that works without any modifications on both synthetic imagery generated by our virtual vision simulator and on real video captured by a physical camera network. This capability, we believe, will help us validate our simulator.

1. wiktorstarzyk@gmail.com
2. adam.domurad@mycampus.uoit.ca
3. faisal.qureshi@uoit.ca

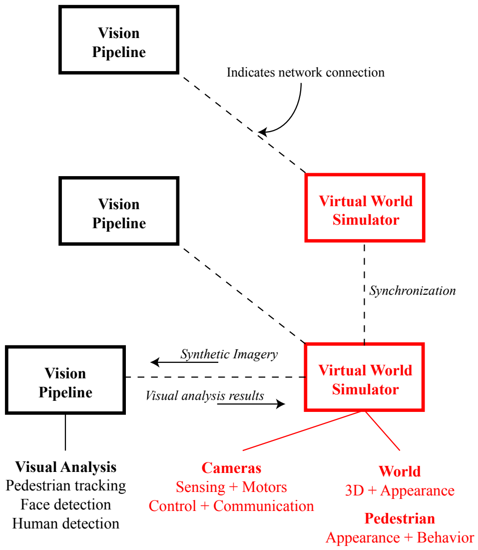


Fig. 2: An overview of our distributed virtual vision simulator.

II. Overview

Our virtual vision simulator consists of two types of modules—1) virtual world (VW) and 2) vision pipeline (VP)—that communicate with each other over the network (Fig. 2). A typical realization of the virtual vision simulator consists of multiple instances of VW and VP modules, spread over a network of computers. VW module records synthetic video of self-animating pedestrians situated in a 3D environment and transmits the synthetic video to one or more VP modules for visual processing. VP modules implement computer vision routines needed to detect, track, and identify pedestrians present in the scene. The results of the visual processing—bounding boxes, pedestrians IDs, etc.—is sent back to VM module which passes this information on to camera control and coordination sub-module. At the moment camera control and coordination module is implemented as a part of the VW module.

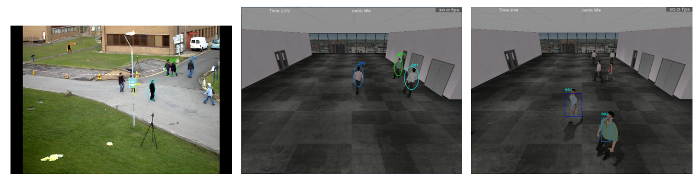
A. Virtual World Module

Our virtual world, shown in Fig. 1, depicts an office building floor complete with panoramic windows showing a view of downtown Toronto as well as realistic looking lighting and shadows. Developed using the Panda3D [3] game engine, it uses the latest graphics technologies to display a "realistic" looking view of a virtual environment. It is populated with self-animating pedestrians that interact with the environment by following a scripted path. The system also models virtual passive and active cameras that can be easily configured and placed anywhere in the environment. We model both passive wide-FOV and active *pan/tilt/zoom* (PTZ). It is straightforward to use different 3D environments in our system.

B. Vision Pipeline Module

We have designed our vision pipeline to work with both real and synthetic imagery without any modifications. This allows us to port any research carried on in our virtual simulator to physical camera networks. Currently, the vision pipeline consists of the following computer vision routines:

- background modeling (MoG/Codebook);
- blob (entry/exit) detection;
- blob tracking (mean-shift/cam-shift/area matching);
- pedestrian tracking (RGB/HSV/Grad/HOG based features); and
- face detection.



(a) Real footage (b) Synthetic footage (c) PTZ tracking

Fig. 3: Our vision pipeline is designed from the ground up to work with both synthetic and real video without any modifications. Consequently, our vision pipeline faithfully mimics the performance of a vision pipeline implemented on a physical camera.

III. Use Cases

So far we have used our virtual vision simulator to develop behavior-based controllers for multi-tasking passive [4] (see Fig. 4) and active cameras and proactive control strategies for active PTZ cameras [5].

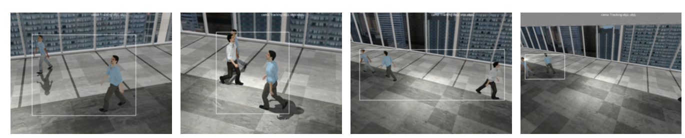


Fig. 4: A multitasking PTZ camera tracking pedestrians as they walk through a hallway.

IV. Summary

We wish to demo our virtual vision simulator to the ICDSC attendees. We seek their feedback in order to improve our simulator and we hope that some of the attendees may find our simulator useful for their own research in the future.

References

[1] F. Z. Qureshi and D. Terzopoulos, "Smart Camera Networks in Virtual Reality," *Proceedings of the IEEE (Special Issue on Smart Cameras)*, vol. 96, no. 10, pp. 1640–1656, Oct. 2008.

[2] W. Shao and D. Terzopoulos, "Autonomous Pedestrians," *Graphical Models*, vol. 69, no. 5-6, pp. 246–274, 2007.

[3] "Panda3d," http://www.panda3d.org, 2011, accessed on 16 June 2011.

[4] W. Starzyk and F. Z. Qureshi, "Multi-tasking smart cameras for intelligent video surveillance systems," in *Proc. 8th IEEE International Conference on Advanced Video and Signal based Surveillance (AVSS 2011)*, 2011, pp. 1–6, to appear.

[5] F. Z. Qureshi and W. Starzyk, "Learning proactive control strategies for ptz cameras," in *Proc. Fifth ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC 2011)*, 2011, pp. 1–6, to appear.