Challenge
While many advances have been made in the realm of computer vision, this is just one aspect of understanding the environment. A richer understanding of the environment can lead to more advanced systems with enhanced capabilities. To advance further, the industry can benefit from tools and technologies to enable tactile sensing and haptic behavior for robots.

Solution
Deep learning algorithms that enable touch as well as vision can create tremendous opportunities for robotics applications. The industry is poised to take advantage of enabling technologies that bring touch to the forefront and are fueled by the latest artificial intelligence (AI) advances.

Background and Project History
Jason Toy, an AI enthusiast, technologist, and founder of Somatic (a company specializing in deep learning image and neuro-linguistic programming models optimized for mobile applications), recently embarked on a project for training AI systems to interact with the environment based on haptic input. Although much work in the AI sector has been done to incorporate visual computing and image processing into solutions, less has been accomplished with touch enablement. Jason’s research focuses on adding sensorimotor neural systems and tactile feedback to robotic systems to expand their mapping of the environment beyond visual imagery to include contours, textures, shapes, hardness, and object recognition by touch.

“I’ve always had a strong interest in artificial intelligence,” Jason said, “and for many years I would devour every book on the mind that I could find. I would purposely study differing points of view in various fields such as physics, philosophy of mind, biology, computer science, mathematics, psychology, linguistics, and so on. One recurrent theme I noted is that intelligence seems to require a body to interact effectively with its environment, not just abstract programs processing text or images.”

SenseNet* is a sensorimotor and touch simulator for teaching artificial intelligence devices how to interact with their environments. Using sensorimotor systems and touch neurons, SenseNet provides a research framework that supports machine-learning researchers and theoretical computational neuroscientists.
“An unknown, but potentially large, fraction of animal and human intelligence is a direct consequence of the perceptual and physical richness of our environment, and is unlikely to arise without it.”

- John Locke, (1632 – 1704), English philosopher

“This idea—of intelligence needing a body—became even more apparent after reading Helen Keller’s book, The Story of My Life,” Jason continued. “You would not even know that she was blind and deaf, if she hadn’t explained that in the book. Her entire life was lived through her hands via touch and she was able to understand the whole world, even concepts that normally require sight. So, to me, if there is a single algorithm for intelligence, it must work with touch as well as vision. Current vision algorithms don’t take into account motion and active movement, whereas with touch, it is a requirement. SenseNet’s goal is to make it easier for the research community to experiment with touch-based machine learning algorithms and devise solutions that involve this technology.”

Jason studied mathematics and computer science at Northeastern University and earned a dual bachelor of science degree in these disciplines. “I’ve always dabbled in neuroscience and artificial intelligence,” he said, “by studying on my own, helping at neuroscience labs, and collaborating with scientists.”

His interest in the entrepreneurial possibilities of technology led to the founding of several startup companies and his latest firm, Somatic, which serves as a consultancy for machine learning and deep learning projects. “I love the ideation and creation of new products,” Jason said, “and bringing them to people to use. I think my experience working in technology startups has enabled me to move fast, test ideas very quickly, and keep my eye on producing things for people to use.”

Jason’s ongoing project, SenseNet: 3D Objects Database and Tactile Simulator, can be used in a reinforcement-learning environment. “The original code,” Jason said, “used OpenAI’s gym* as the base. Any code written for gym can be used with little to no further tweaking. In many instances, you can just replace gym with sensenet and everything will work.”

Figure 1 shows Jason presenting his project at an ICML event in which Intel® AI Developer Program participated.

Immediate applications of the work being accomplished in the SenseNet project include developing robotic hands for use in factories and distribution centers to perform bin packing, parts retrieval, order fulfillment, and sorting. Any application that requires robotics to handle real-world objects sensitively—such as preparing food, performing household tasks, and assembling components—would be effective use of the technology.

The algorithms developed during the project could also spur future research and work well in instances in which touch-capable robots offer benefits.

Enabling Technologies

“I used Intel’s Reinforcement Learning Coach to help accelerate training and testing of many state-of-the-art reinforcement learning algorithms,” Jason said. “Coach is a machine learning test framework that includes implementations of state-of-the-art reinforcement learning algorithms that you can run on your environments. SenseNet’s API interface is practically the same as OpenAI’s and so integration with Coach was very easy.”

Functioning within a Python* environment, Coach lets developers model the interaction between an agent and the environment, as shown in Figure 2. By combining various building blocks and providing visualization tools to dynamically display training and test results, Coach makes the training process more efficient, as well as supporting testing of the agent on multiple environments.

Figure 1. SenseNet* presentation at the Vancouver International Conference of Machine Learning 2018.

“Our overarching goal in releasing SenseNet is to create that initial spark of research and exploration in sensorimotor systems and tactile feedback that ImageNet* did for the computer vision field.”

– Jason Toy, Somatic founder and AI researcher
Coach supports a variety of environments. Agent tests can be performed for specialized applications, including robotics, computer gaming, and more. The advanced visualization tools, based on data collected during the training sequences, can be readily accessed through the Coach dashboard and used to debug and optimize the agent being trained. Figure 3 shows an example of the types of information provided through the Coach dashboard.

Figure 2. Modeling an agent by combining building blocks.

Figure 3. Dynamic display available through the Coach dashboard.
"Intel’s Coach definitely helped speed up the process in bootstrapping testing of new algorithms,” Jason noted. “Without it, we would have been slowed down a lot, testing whether our algorithms were implemented properly.”

Part of the value of working with Intel, from Jason’s perspective, was having access to AI teams willing to share their expertise and to explore different ideas and approaches to daily challenges.

In terms of advice to other developers, Jason said, “Don’t be afraid to go against the norm. Most of the deep learning craze is centered around convolutional neural networks (CNNs) and computer vision, since that is where the most gains have occurred.” Other less-explored areas offer insights and sometimes breakthroughs in AI, he feels, and these less popular avenues can lead in promising directions and unlock opportunities for advances.

“As also,” Jason continued, “don’t just study artificial intelligence from the point of view of mathematics and computer science. Look at other fields such as computational neuroscience and cognitive science.”

Deep Reinforcement Learning and Robotics

Reinforcement learning (RL), a branch of machine learning that draws from both supervised and unsupervised learning techniques, relies on a system of rewards based on monitored interactions, finding better ways to improve results iteratively. The working model used in a typical reinforcement learning algorithm is progressively improved over time by sensing input signals, deciding on an appropriate action, and then assessing the results of the action in terms of a reward definition. Within this model, the RL algorithm works by itself without direct supervision, discovering techniques to optimize the rewards obtained.

However, RL cannot be considered fully unsupervised, because the definition of the reward is determined before the learning process begins and all of the responses taken and the results measured are based on that reward definition.

Reinforcement learning has promise in the field of robotics, offering mechanisms that could enable autonomous robots to master certain independent behaviors with minimal human intervention. Evaluations of deep reinforcement learning techniques indicate that it is possible to use simulation to develop dexterous 3D manipulation skills without having to manually create the representations. One paper that explains how reinforcement learning can be used for simulated robotic interactions is titled Deep Reinforcement Learning for Robotic Manipulation with Asynchronous Off-Policy Updates. Another useful reference on this topic is Data-efficient Deep Reinforcement Learning for Dexterous Manipulation.

Using the SenseNet* Dataset

As a part of this project, Jason has been building an open source dataset of shapes, most of which can be 3D printed, as well as a touch simulator so that other AI researchers can accelerate their own project work in this area. The SenseNet repository on GitHub* provides numerous resources beyond the 3D object dataset, including training examples, classification tests, benchmarks, Python* code samples, and more. Figure 4 shows examples of some of the shapes included in the dataset.

![Figure 4. Examples of SenseNet* 3D objects.](image)

The extensive dataset is a valuable research tool, but it is also made more useful through the addition of a simulator included with SenseNet that lets researchers load and manipulate the objects. “We have built a layer upon the Bullet* physics engine,” Jason explained. “Bullet is a widely used physics engine in games, movies, and—most recently—robotics and machine learning research. It is a real-time physics engine that simulates soft and rigid bodies, collision detection, and gravity. We include a robotic hand called the MPL that allows for a full range of motion in the fingers and we have embedded a touch sensor on the tip of the index finger that allows the hand to simulate touch.” See Figure 5 for some of the supported hand gestures that are available using MPL.

![Figure 5. Robotic hand gestures available in SenseNet*.](image)

SenseNet—and the collection of resources that supports it—removes many of the obstacles that researchers may encounter when first getting involved with haptic experimentation.

Figure 6 shows an example of objects that have been 3D printed from the database.
AI is Expanding the Boundaries of Robotics

Through the design and development of specialized chips, research, educational outreach, and industry partnerships, Intel is accelerating the progress of artificial intelligence (AI) to solve difficult challenges in medicine, manufacturing, agriculture, scientific research, robotics, and other industry sectors. Intel works closely with policymakers, educational institutions, and enterprises of all kinds to uncover and advance solutions that address major challenges in the sciences.

Another example of haptic research: An ongoing project being conducted by MIT’s Computer Science and Artificial Intelligence Laboratory is refining a landmark sensor technology, dubbed GelSight, that equips robots with a sense of touch. The research team, led by Ted Adelson, employs rubbery membranes affixed over tiny cameras at the robot’s fingertips. Through physical contact and pressure, GelSight maps the surfaces of objects in 3D through touch, determining the degree of hardness that any object exhibits. This capability is considered a prerequisite for training household or industrial robots to manipulate everyday objects and tools.

Wenzhen Yuan, co-author of the two papers on this technology that were presented at the International Conference on Robotics and Automation in June 2017, developed the experimental process, fashioning objects within confectionary molds. Each object had a uniform shape but a varying degree of hardness, as measured using a standard industrial scale.

Data was collected by pressing the GelSight sensor against the objects and visually mapping the contact pattern over time. The deformation of the object at five individual stages was recorded and this data was fed into a neural network to identify correlations between contact patterns and hardness measurements. In an informal experiment, the accuracy of this technique was validated by comparing human assessments of the degree of hardness of various fruits and vegetables. The robot equipped with the GelSight sensor detected the same degree of hardness in each case as the human participants in the experiments.

“Vision is simply not suited to the nature of the problem: Grasping tasks are a matter of contact and forces, which cannot be monitored by vision. At best, vision can inform the robot about finger configurations that are likely to succeed, but in the end a robot needs tactile information.”

– Vincent Duchaine, professor at École de Technologie Supérieure (ÉTS) in Montreal, Canada

Figure 6. 3D-printed SenseNet* objects for research experimentation.

Figure 7. 3D printer and heated build plate for 3D printing SenseNet* objects.
“Software is finally catching up with the capabilities of our sensors. Machine learning algorithms inspired by innovations in deep learning and computer vision can process the rich sensory data from sensors such as the GelSight to deduce object properties. In the future, we will see these kinds of learning methods incorporated into end-to-end trained manipulation skills, which will make our robots more dexterous and capable, and maybe help us understand something about our own sense of touch and motor control.”

– Sergey Levine, assistant professor of Electrical Engineering and Computer Science, University of California at Berkeley

The Intel® AI technologies used in this implementation included:

**Reinforcement Learning Coach**: Provides an open source research framework for training and evaluating RL agents by harnessing the power of multi-core CPU processing to achieve state-of-the-art results.

**Framework Optimization**: Achieve faster training of deep neural networks on a robust scalable infrastructure.

**Intel® Xeon® Scalable processors**: Tackle AI challenges with a compute architecture optimized for a broad range of AI workloads, including deep learning.

**OpenVINO™ toolkit**: Make your vision a reality on Intel® platforms—from smart cameras and video surveillance to robotics, transportation, and more.

**Intel® Distribution for Python**: Supercharge applications and speed up core computational packages with this performance-oriented distribution.

**Intel® Data Analytics Acceleration Library (Intel® DAAL)**: Boost machine learning and data analytics performance with this easy-to-use library.

**Intel® Math Kernel Library (Intel® MKL)**: Accelerate math processing routines, increase application performance, and reduce development time.

For more information, visit the [portfolio page](#).

**RESOURCES**

Intel® AI Developer Program [software.intel.com/ai](http://software.intel.com/ai)

Inside Artificial Intelligence – Next-level computing powered by Intel AI [ai.intel.com/](http://ai.intel.com/)

Intel® AI DevCloud – Free cloud compute for Intel AI Developer Program members [software.intel.com/ai/devcloud](http://software.intel.com/ai/devcloud)

Intel® Developer Mesh SenseNet project [devmesh.intel.com/projects/sensenet](http://devmesh.intel.com/projects/sensenet)

SenseNet repository – Examples, images, and access to a dataset of thousands of objects for simulator manipulation [github.com/jtoy/sensenet](http://github.com/jtoy/sensenet)


PyBullet* Quickstart Guide – Physics simulation for games, visual effect, and robotics [https://pybullet.org/wordpress/](https://pybullet.org/wordpress/)

Intel® Software Innovator Program - Supports innovative, independent developers [software.intel.com/intel-software-innovators/overview](http://software.intel.com/intel-software-innovators/overview)

Intel® Optimization for Caffe* [software.intel.com/ai/frameworks/caffe](http://software.intel.com/ai/frameworks/caffe)
SUCCESS STORY | Deep Learning Brings Touch to Robots

1 https://spectrum.ieee.org/automaton/robotics/robotics-hardware/why-tactile-intelligence-is-the-future-of-robotic-grasping

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps.

The products and services described may contain defects or errors known as errata which may cause deviations from published specifications. Current characterized errata are available on request.

Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or by visiting www.intel.com/design/literature.htm.

Intel, the Intel logo, OpenVINO, and Xeon are trademarks of Intel Corporation in the U.S. and/or other countries.

*Other names and brands may be claimed as the property of others.

© 2018 Intel Corporation Printed in USA 0618/KP/MESH/PDF 337325–001US