



**AI ON  
INTEL**

**从数据中心到边缘 —  
使用英特尔架构的人工  
智能解决方案**



# 法律信息

这些材料仅供学习参考，需遵守以下链接中的 CC\_BY\_NC\_ND 4.0 许可：<https://creativecommons.org/licenses/by-nc-nd/4.0/>

英特尔技术的特性和优势取决于系统配置，可能需要支持的硬件、软件或服务激活。实际性能可能因系统配置的不同而有所差异。没有任何产品或组件能够保证绝对安全。请咨询您的系统制造商或零售商，也可登录 [intel.com](https://intel.com) 获取更多信息。

本文件不构成对任何知识产权的授权，包括明示的、暗示的，也无论是基于禁止反言的原则或其他。

本文包含尚处于开发阶段的产品、服务和/或流程的信息。此处提供的所有信息如有更改，恕不另行通知。请联系您的英特尔代表，了解最新的预测、时间表、规格和路线图。

优化声明：英特尔的编译器针对非英特尔微处理器的优化程度可能与英特尔微处理器相同（或不同）。这些优化包括 SSE2、SSE3 和 SSSE3 指令集以及其它优化。对于在非英特尔制造的微处理器上进行的优化，英特尔不对相应的可用性、功能或有效性提供担保。此产品中依赖于处理器的优化仅适用于英特尔微处理器。某些不是专门面向英特尔微体系结构的优化保留专供英特尔微处理器使用。请参阅相应的产品用户和参考指南，以了解关于本通知涉及的特定指令集的更多信息。

在性能测试过程中使用的软件及工作负载可能仅针对英特尔微处理器进行了性能优化。SYSmark\* 和 MobileMark\* 等性能测试使用特定的计算机系统、组件、软件、操作和功能进行测量。上述任何要素的变动都有可能导致测试结果的变化。请参考其他信息及性能测试（包括结合其他产品使用时的运行性能）以对目标产品进行全面评估。

英特尔、英特尔标识、Arria、Myriad、Atom、凌动、Xeon、至强、Core、酷睿、Movidius、neon、Stratix、OpenCL、Celeron、赛扬、Phi、融核、VTune、Iris、锐炬、OpenVINO、Nervana、Nauta 和 nGraph 是英特尔公司在美国和/或其他国家的商标。

\*其他的名称和品牌可能是其他所有者的资产。

© 2019 年英特尔公司版权所有。所有权保留



# 数据集引用

多样化的大型数据集可改善汽车制造和型号识别

F. Tafazzoli, K. Nishiyama 和 H. Frigui

2017 年 IEEE 计算机视觉和模式识别大会 (CVPR) 会议纪要 ([http://vmmdb.cecsresearch.org/papers/VMMR\\_TSWC.pdf](http://vmmdb.cecsresearch.org/papers/VMMR_TSWC.pdf)).





# 课程结业证书

- 您可以选择在完成课程测验时获得英特尔® 人工智能课程结业证书。
- 开始测验前，您可能需要禁用广告拦截工具。（Ghostery、uBlock、AdGuard 等）





# 学习目标

使用英特尔硬件和软件产品组合，展示数据科学流程

- 在实践中了解如何构建深度学习模型和部署至边缘
  - 使用企业图像分类问题
  - 对 VMNR 数据集实施探索性数据分析
  - 选择框架和网络
  - 训练模型 — 获取训练后网络的图形和权重
  - 在 CPU、集成显卡和英特尔® Movidius™ 神经计算棒上部署该模型



# 训练概述

## 1. 英特尔的人工智能产品组合

- 硬件：从训练到推理，重点介绍第二代英特尔® 至强® 可扩展处理器
- 软件：针对英特尔架构优化的框架、库和工具
- 社区资源：英特尔开发人员专区资源

## 2. 探索性数据分析

- 获取数据集
- 以可视化方式探索数据，了解分布情况
- 数据精简和解决不平衡问题

## 3. 训练模型

- 基础设施：英特尔® AI DevCloud、Amazon Web Services\*、Google Compute Engine\*、Microsoft Azure\*
- 流程：准备数据集并对其进行可视化，学习如何选择合适的框架和模型，超参数调优，训练及验证

## 4. 模型分析

- 查看分数
- 比较结果
- 超参数调优
- 选择理想的模型或者重新训练

## 5. 部署到边缘/推理

- 英特尔® OpenVINO™ 工具套件简介 — 功能和优势
- 使用模式
- 模型优化器 — 优化模型，并为预构建和定制模型生成独立于硬件的中间表示 (IR) 文件
- 推理引擎 — 部署到 CPU、集成 GPU、FPGA 和英特尔® Movidius™ 神经计算棒



# 前提条件

- 基本了解人工智能原理、机器学习和深度学习
- Python 的编程体验
- 了解不同的框架 — Tensorflow\*、Caffe\* 等
- 下面列出了可以帮助您快速入手的一些教程
  - 人工智能简介 (<https://software.intel.com/en-us/ai/courses/artificial-intelligence>)
  - 机器学习 (<https://software.intel.com/en-us/ai/courses/machine-learning>)
  - 深度学习 (<https://software.intel.com/en-us/ai/courses/deep-learning>)
  - Tensorflow\* 的应用深度学习 (<https://software.intel.com/en-us/ai/courses/tensorflow>)





# 英特尔人工智能 产品组合





Outline of 30 minute section





So what's driving this AI surge?

Data, for one. In 2019, the average internet user will generate ~25GB of IP traffic **per month**. By comparison, in a **single day**, a smart car will generate 2X that amount of data (50 GB), a smart hospital will generate 120X (3TB or 3,000 GB), a plane will generate 1,600X (40TB or 40,000 GB), a smart factory will generate 40,000X (1PB or 1,000,000 GB), and a city safety system will generate 800,000X (50PB or 50,000,000 GB). And we're only talking about 2019! When you consider that there will be 3X more smart connected devices than the global population by 2022, a growth from 17.7 billion networked devices in 2019 to 28.5 billion in 2022, the quantity of data generated is difficult to fathom. This data contains a treasure trove of valuable insights, in business, operations and security that we really want to extract, and in order to do that efficiently we need tools like analytics and AI by our side.

And when you think about analyzing troves of data, the first thing that may come to mind is “data analytics”, the longstanding but constantly evolving science that companies leverage for insight, innovation, and competitive advantage. Analytics has changed a lot over the years, but continues to advance through

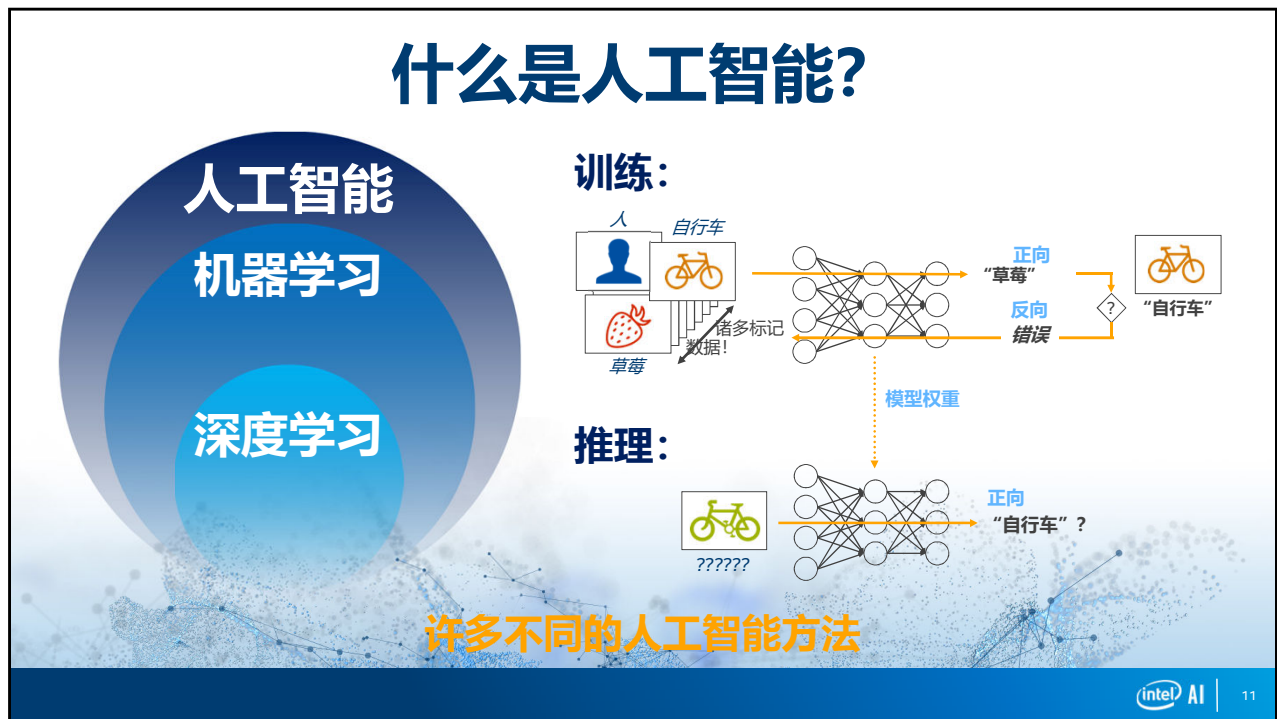


‘more or less’ five stages of increasing scale & maturity:

- Descriptive & Diagnostic analytics, sometimes called “operational analytics” , help us understand what happened and why.
- Predictive, Prescriptive & Cognitive analytics, sometimes called “advanced analytics” , help us predict and plan for the future.

AI is its own category, applied to all phases of the analytics pipeline (especially more advanced analytics), and a vital tool for reaching higher maturity & scale data analytics. And with the recent breakthroughs in computation performance and an in-pouring of innovation into the A+AI realm, we now have the tools to extract valuable insights from troves of data.





So, what is AI, exactly? Well, we’ re a long way from how Artificial Intelligence (AI for short) is portrayed in science fiction movies. The definition continues to evolve, but fundamentally, **AI** is the ability of machines to learn from experience, without explicit programming, in order to perform functions typically associated with the human mind. **There’ s no one-size fits all approach to AI, so it’ s helpful to explore some of the more prominent approaches to AI.**

One such leading approach is **machine learning**, a category includes algorithms that improve with exposure to more data over time, and there are countless such algorithms that perform functions like regression, classification, clustering, decision trees, extrapolation, and more.

A fast-growing subset within the machine learning category is **deep learning**. This approach uses layered neural networks that learn from vast amounts of data to solve problems that are difficult to reverse engineer, such as computer vision, speech recognition, and many more. With deep learning, we avoid feature some of the ‘reverse engineering’ required with traditional machine learning algorithms, instead letting the neural network automatically adjust and adapt to every new piece of training data.

Now, let’ s explore the difference between the two key stages of deep learning: training



and inference.

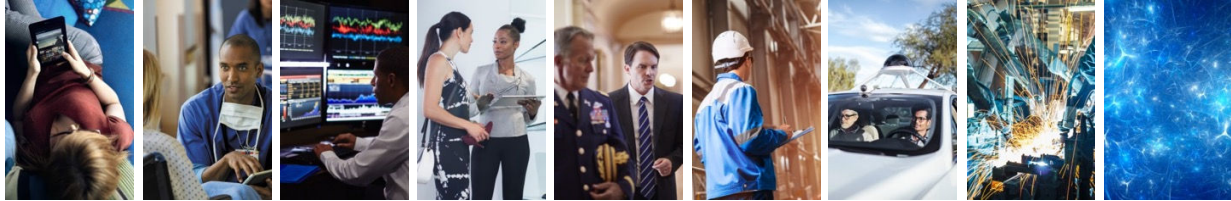
In the example shown here, the job of the deep neural network is to classify a picture into one of three different categories – a person, a bicycle, or a strawberry.

First, labeled image (e.g. picture of a bicycle labeled as “bicycle” ) is input into the network in a “forward pass” and the untrained network predicts that the bicycle is a strawberry, which is an error. Next, in the “backward pass” , the error propagates back through the network and the weights (i.e., the interconnections between the artificial neurons) are updated to account for the error. Once these updates have been made, the next time the same image is passed into the network, it will be more likely to predict that it’ s a bicycle. Over billions upon billions of iterations like this example, you end up with a “trained” neural network that can accurately identify a given input image. When you’ re satisfied with the trained accuracy of your neural network, the model weights are frozen, and the trained model can be used for inference.

Inference, in this example, is the process of feeding an unknown image into the trained neural network and allowing it to “infer” what’ s in that image. If you did a good job training the model weights, it should predict “bicycle” for an image of a bicycle. Inference is really just the “forward pass” portion of the training phase, but while training is dense compute intensive and typically done in the data center, inference can take place there or even in a smart car or on a smartphone. The compute demands for inferencing really depend on the use case, and vary significantly in throughput, latency, power and size. So while you could use the same processor for inference as you do for training, it often makes sense to use a different more efficient approach.



# 人工智能将会革新



## 消费电子    医疗    金融    零售    政府    能源    运输    工业    其他

<ul style="list-style-type: none"> <li>智能助理</li> <li>聊天机器人</li> <li>搜索</li> <li>个性化</li> <li>增强现实</li> <li>机器人</li> </ul>	<ul style="list-style-type: none"> <li>增强诊断</li> <li>药物发现</li> <li>患者护理</li> <li>研究</li> <li>助听器</li> </ul>	<ul style="list-style-type: none"> <li>算法交易</li> <li>欺诈检测</li> <li>研究</li> <li>个人财务</li> <li>降低风险</li> </ul>	<ul style="list-style-type: none"> <li>支持</li> <li>体验</li> <li>市场营销</li> <li>营销</li> <li>客户忠诚度</li> <li>供应链</li> <li>安全性</li> </ul>	<ul style="list-style-type: none"> <li>国防</li> <li>数据洞察</li> <li>安全和保障</li> <li>居民互动</li> <li>智慧城市</li> </ul>	<ul style="list-style-type: none"> <li>石油和天然气勘探</li> <li>智能电网</li> <li>改善运营</li> <li>节能</li> </ul>	<ul style="list-style-type: none"> <li>车载体验</li> <li>自动驾驶</li> <li>航空</li> <li>运输</li> <li>搜索与救援</li> </ul>	<ul style="list-style-type: none"> <li>工厂自动化</li> <li>预测性维护</li> <li>精准农业</li> <li>工业现场自动化</li> </ul>	<ul style="list-style-type: none"> <li>广告</li> <li>教育</li> <li>游戏</li> <li>专业与 IT 服务</li> <li>电信/媒体</li> <li>体育</li> </ul>
---	---	--	---	---	--	---	---	--

来源：英特尔预测



Which industries are the earliest adopters of AI? Generally, those segments with clear use cases, high purchasing power, and high rewards for making decisions quickly and/or more accurately will adopt AI fastest. Here are the segments that we believe will lead AI through 2020, ordered roughly by market opportunity (earliest at left).

----- BACKUP -----

## 消费电子

- Smart Assistants – personal assistant that anticipates, optimizes, automates daily life (e.g. Amazon Alexa, Apple Siri, Google Assistant, Microsoft Cortana, Facebook Jarvis home automation, X.ai virtual assistant Amy)
- Chatbots – 24/7/365 no waiting access to an informative or helpful agent (e.g. WeChat, Bank of America, Uber, Pizza Hut, Alaska Airlines, Amtrak, etc.)
- Search – ability to more intelligently search more data types including image, video, context, etc (e.g. Improved Google search, Google Photos, ReSnap)
- Personalization – ability to automatically adjust content/recommendations to suit individuals (e.g. Entefy, Netflix recommendation engine, Amazon personalized shopping recommendations)
- Augmented Reality – overlay information on our field of view in real-time to identify interesting or undesirable things (e.g. Intel Project Alloy, Google Translate using smartphone)



camera)

- Robots – personal robots that are able to perform household, yard, or other chores (e.g. Jibo robot for day-to-day functions, Roomba follow-ons)

### **Health** (SME:Kristina Kermanshahche, Ketan Paranjape)

- Enhanced Diagnosis – a tool for doctors to augment their own diagnosis with more data, experience, precision and accuracy (e.g. radiology image analysis, Journal of American Medicine Association paper on retina scan for diabetic retinopathy, skin lesion classification to recognize melanoma with 98% accuracy, medical history scraping, treatment outcome prediction)
- Drug Discovery – computational drug discovery that intelligently hones in on the most promising treatments (e.g. speeding pharma drug development)
- Patient Care – machines that aid with monitoring, treatment, and/or recovery of patients (e.g. visual patient monitoring, autonomous robotic surgery, friendly medication and/or physical therapy robots)
- Research – instantly sifting through hundreds of new research papers and clinical trials that are published each day to make new connections (e.g. AI at University of North Carolina’s Lineberger Comprehensive Cancer Center)
- Sensory Aids – filling in for various senses that are absent or challenged (e.g. visual aid, audio aid)

### **Finance** (SME:Robert Geva)

- Algorithmic Trading – augment rule-based algorithmic trading models and data sources using AI (e.g. Kensho analysis of myriad data to predict stock movement)
- Fraud Detection – ability to identify fraudulent transactions and/or claims (e.g. USAA identifies insurance fraud)
- Research – ability to intelligently assemble, parse, and extract meaning from troves of data that influence asset prices (e.g. Quid, FSI firm reducing time to insight for portfolio managers through smart knowledge management system)
- Personal Finance – smarter recommendations, lower risk lending, greater efficiency (e.g. active portfolio recommendations, quickly parsing more data before issuing loan, automatic reading of check scans, etc.)
- Risk Mitigation – detect risk factors and/or reduce the burden of regulation and minimize errors through automated compliance (e.g. IBM+Promontory Financial Group using natural language processing to detect excursions)

### **Retail** (SME:Janet Kerby, Chris Hunt)

- Support – bots providing shopping, ordering and support in lifelike interaction (e.g. My Starbucks Barista, KLM Dutch Airline customer support via social media, Nieman Marcus visual search, Pizza Hut order pizza via bot, Adobe Digital’s digital mirror that recommends clothes, intelligent phone menu routing based on NLP, ViSenze recommending similar items based on image, Adobe Digital’s digital mirror that recommends clothes)
- Experience – deliver winning consumer experiences in-store (e.g. Amazon Go checkout-free grocery store, Macy’s mobile shopping assistant, Lowes Lowebots that roam stores)



answering simple questions and tracking inventory)

- Marketing – precision marketing to consumers, promoting products and services how and where they want to hear (e.g. North Face “Expert Personal Shopper” on website)
- Merchandising – better planning through accelerated and expanded insight into consumer buying patterns (e.g. Stitch Fix virtual styling, Skechers.com analyzing clicks in real-time to bring similar catalog items forward, Wal-mart pairing products that sell together, Cosabella evolutionary website tweaks)
- Loyalty – transform the consumer experience through segmentation (e.g. Under Armour health app that constantly collects user data to deliver personalized fitness recommendations)
- Supply Chain – optimize the supply chain and inventory management for efficiency and innovate new business models (e.g. OnProcess technology’s use of predictive analytics for inventory management)
- Security – improve security of all consumer and business digital assets, such as real-time shoplifting/lifter detection, multi-factor identity verification, data breach detection (e.g. Mastercard pay with your face, Walmart facial recognition to catch shoplifters)

#### **Government** (SME:Harris Joyce)

- Defense – drones, connected soldiers, defense strategy (e.g. military/surveillance drones, autonomous rescue vehicles, augmented connected soldier, real-time threat assessment and strategy recommendation)
- Data Insights – analyze massive amounts of data to identify opportunities/inefficiencies in bureaucracy, cybersecurity threats and more, to ultimately implement better systems and policies (e.g. MIT AI that detects cyber security threats)
- Crime Prevention using AI to predict and help recover from disasters thanks to ability to quickly process large amounts of unstructured data and optimize limited resources (e.g. 1Concern, BlueLineGrid)
- Safety & Security – crowd analytics, behavioral/sentiment analytics, social media analytics, face/vehicle recognition, online identity recognition, real-time video analytics, using AI to predict and help recover from disasters thanks to ability to quickly process large amounts of unstructured data and optimize limited resources (e.g. police analyzing social media to adjust police presence, license plate readers in police cars, 1Concern, BlueLineGrid)
- Resident Engagement – new tools to facilitate citizen engagement like chatbots, at-risk citizen identification, (e.g. Amelia chatbot in North London Enfield council, North Carolina chatbot to help state employees with IT inquiries)
- Smarter Cities – traffic/pedestrian management, lighting management, weather management, energy conservation, services analytics (e.g. San Francisco and Pittsburgh using sensors and AI to optimize traffic flow)

#### **Energy** (SME:Noe Garcia, Tonya Cosby)

- Oil & Gas Exploration – automated geophysical feature detection (e.g. oil & gas producers using AI to augment traditional modeling & simulation)
- Smart Grid – predictive and real-time intelligent generation, allocation, and storage of power to meet variable demand (e.g. GridSense, SoloGrid)
- Operational Improvement – safety and efficiency improvements through predictive and/or insightful AI (e.g. GE Oil and Gas using predictive analytics and AI to predict and preempt



potential operational problems)

- Conservation – intelligent buildings, computing and appliances that reduce power consumption and are more efficient than producing another kWh of electricity (e.g. Google DeepMind datacenter energy reductions)

### **Transport** (SME:Len Klebba)

- Automated Cars – autonomous cars driving on the roadways (e.g. BMW, Google, Uber, many others)
- Automated Trucking – autonomous trucks driving on the roadways (e.g. Daimler)
- Aerospace – autonomous planes and other aerial vehicles (e.g. Boeing’ s evolution of autopilot and drones)
- Shipping – autonomous package delivery via drone or other vehicle (e.g. Amazon package delivery drone)
- Search & Rescue – ability to deploy autonomous robot to search and rescue victims in potentially hazardous environments (e.g. war casualty extraction, miner rescue, firefighting, avalanche rescue)

### **Industrial** (SME:Mary Bunzel, Esther Baldwin)

- Factory Automation – highly-productive, efficient and safe factories with robots that can see, hear and adapt to their environment to produce goods with incredible quality and speed (e.g. assembly line)
- Predictive Maintenance – ability to detect patterns that indicate the likelihood of an upcoming fault that would require maintenance (e.g. airline being able to adjust schedule to perform preventive maintenance before a failure)
- Precision Agriculture – ability to deliver the precise amount of water, nutrients, sunlight, weed killer, etc to a particular crop or individual plant (e.g. farmer using visual weed search to zap only weeds with RoundUp, automated sorting of produce for market)
- Field Automation – ability to automate heavy equipment beyond the factory walls (e.g. mining, excavation, construction, road repair)

### **其他**

- Advertising – interactive ads, adaptive ads, personalized ads, real-time ads (e.g. AdBrain, MetaMarkets, Proximic, RocketFuel)
- Education – virtual mentors, foreign language instruction, automated study sheets, personalized assignments, cheating detection, deliberate practice, machine-to-machine instruction (e.g. Intelligent Tutor Systems, Content Technologies Inc, PR2 robot from Cornell)
- Gaming – dynamic and interactive video game experiences (e.g. Xbox Kinect, Playstation Eye, Wii)
- Professional & IT Services – sales, marketing, legal research, accounting/tax, assisted counseling, customized IT recommendations (e.g. Pinsent Masons law firm that emulates human decision-making, Salesforce use of AI)
- Telco/Media – customized content/ads, network optimization, quality of service, mobile/home security (e.g. media company customizing tv show recommendations and ads, network operator ensuring efficient and high-quality delivery/repair, wireless company using



multi-factor security)

- Sports – intelligent analytics for injury prevention and betting (e.g. Kinduct injury prevention, Microsoft Cortana predicting football games)

Here is an even broader list of industries that will be impacted by AI:Advertising, Aerospace, Agriculture, Automotive, Building Automation, Business, Education, Fashion, Finance, Gaming, Government, Healthcare, IT, Investment, Legal, Life Sciences, Logistics, Manufacturing, Media & Entertainment, Oil/Gas/Mining, Real Estate, Retail, Sports & Fitness, Telecommunications, Transportation

来源： Intel forecast (IDC, GII Research, Tractica, Technavio, Market Research Store, Allied Market Research, BCC Research)



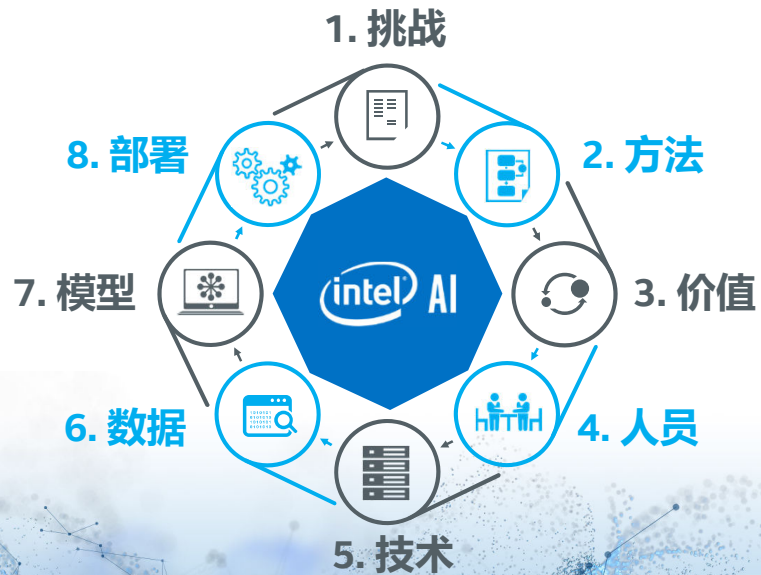


Outline of 30 minute section



# 人工智能之旅

与英特尔携手合作，加速  
您的人工智能之旅



Before we venture any further, it's important to understand that implementing AI in your organization will be a journey, and to think about which technology partner can help you accelerate each step to take you full circle. **In the next four slides, we'll walk through an Intel AI case study to illustrate this journey.**





Outline of 30 minute section



# 打破人工智能理论与现实之间的界限



与英特尔合作，加速您的人工智能之旅



The Intel AI commitment to our customers is simple: we're here to help them break barriers between AI theory and reality. With Intel AI, our customers can simplify AI, choose any approach, tame their data deluge, speed up development, deploy AI anywhere, and scale with confidence. There's no other company on the planet that brings these unique capabilities together to accelerate AI from start to finish.

At the heart of these capabilities, are three things:

First is our AI **hardware** portfolio. Intel brings unprecedented AI hardware choice, from Intel® Xeon® Scalable processors optimized for faster deep learning, to Intel® Movidius™ VPUs for leading on-device computer vision inference, to Intel® FPGAs for real-time inference, to forthcoming ASICs built from the ground up to accelerate deep learning, and more. Through these compute engines and our innovative memory, storage & connectivity technologies, we're helping our customers move, store and process data for AI.

Second is our AI **software** stack, which is inextricable from hardware. From research to deployment, Intel is contributing open software & tools to speed up AI, including optimizations for the most popular open-source deep learning frameworks and topologies to get the most out of our hardware.

Third is our AI **community** which brings it all together. Intel's community helps



customers truly move from data strategy to enterprise-scale AI deployment, through AI direct engagements, market-ready solutions, reference designs, and many more offerings across all industries.

For more information about all these and more, visit [www.intel.ai](http://www.intel.ai)



# 硬件





With Intel AI, you can deploy AI anywhere with unprecedented hardware choice.

As you can infer (pun intended), each AI use case has very different requirements in terms of compute, power, size/form factor, latency, cost, resilience, etc. It's helpful to break these requirements down into a few buckets:

- On the top-left is the **device** category, where end point uses with lower power interactive technology reside such as personal computers, cameras, smart speakers, drones, robots and more. For this category, the Intel® Atom™ and Intel® Core™ processors are frequently the host processor, but we're seeing a growing demand in this space for domain specific inference SOC's tailored to individual applications
- For internet of things sensors (IOT) in security, home, retail, industrial and many more verticals, high performance with very low power is crucial. For vision & inference applications in drones and cameras for example, the Intel® Movidius™ Vision Processing Units (VPU) deliver high quality image recognition in a <1 watt power envelope. You can experience this platform through the Intel® Movidius™ Neural Compute Stick NCS (<https://developer.movidius.com>). Similarly, for speech recognition in smart speakers and robots for example, the combination of the Intel® Atom™ processor with the Intel GNA (Gaussian Neural Accelerator), you can enable always-on listening using only milliwatts of power. You can experience this platform through the Intel speech enabling developer kit



(<https://software.intel.com/en-us/iot/speech-enabling-dev-kit>).

- For self-driving vehicles, Intel® Mobileye™ technology is your autonomous driving solution – it’s a comprehensive self-driving vehicle platform that’s been in development for years. For transportation-as-a-service oriented companies that want control over their IP and access to the bare silicon, the Intel® Nervana™ Neural Network Processor (codename Spring Hill; coming Q4’ 19) for inference is the best solution.
- For personal computing, including desktops, laptops, convertibles, tablets, smartphones and more, Intel is combining several of our dedicated accelerators (Movidius VPU, GNA) with our CPU technology (Atom, Core) and integrated processor graphics (Intel® Iris graphics) to deliver game-changing display, video/vision, AR/VR, speech and gesture capabilities.

• On the left-middle is the **edge**, which could be a small distributed cluster located at a company’s factories around the world, an aggregation point like a network video recorder (NVR), a complex system like a car or MRI (magnetic resonance imaging) machine, or even just a few servers or workstations acting as gateway devices. In other words, the “edge” is a broad category for localized compute. For the most part, most customers are doing all their deep learning inferencing on Xeon/CPU, unless they’re consistently doing a tremendous amount of it and/or have specific use case requirements, which drives demand for general purpose acceleration. Even in that case, for customers who run into problems running inference on CPU, upgrading to the latest generation Xeon and utilizing the latest Intel-optimized deep learning software (frameworks & topologies) can help meet their demands.

- For dedicated inference applications, the Intel® Nervana™ Neural Network Processor for inference (codename Spring Hill; coming Q4’ 19) will likely be the most efficient solution
- For vision & inference workloads with higher performance/watt requirements, the Intel® Movidius™ vision processing unit (VPU) is a great option, available as a PCIe add-in card called the “Intel® Vision Accelerator Design with Intel® Movidius™ Myriad™ X VPU”
- For streaming latency-bound workloads with “real-time” inference demands, particularly in media & vision, the highly-flexible Intel® Arria® 10 FPGA is another option, available as a PCIe add-in card called the “Intel® Vision Accelerator Design with Intel® Arria® 10 FPGA”

• Finally, on the left-bottom is **multi-cloud**, which consists of the largest ‘hyperscale’ deployments such as public clouds (AWS, GCP, etc), communication service providers, government labs, academic clusters, large enterprise IT (private and/or hybrid cloud) and more. For the most part, most customers are running their deep learning inferencing and training on Xeon/CPU, unless they’re consistently doing a tremendous amount of it, which drives demand for general purpose acceleration. Even in that case, for customers who run into problems running on CPU, upgrading to the latest generation Xeon and utilizing the latest Intel-optimized deep learning software (frameworks & topologies) can help meet their demands.

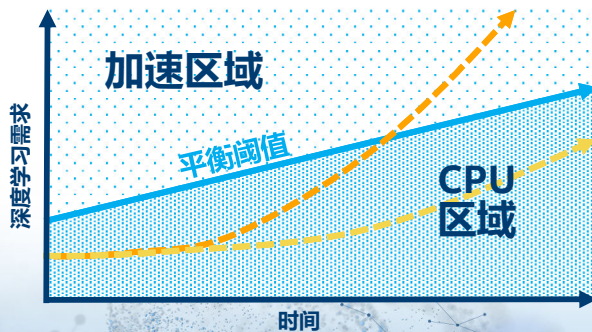


- For dedicated deep learning training environments, like those where that workload is persistent and accounts for a large share of all compute cycles, the Intel® Nervana™ Neural Network Processor (NNP) is your purpose-built AI accelerator solution with multi-model and multi-user support – coming in 2019.
- For customers with memory-bandwidth bound (e.g. RNN/recurrent neural networks) and/or flexible acceleration needs, the Intel® Stratix® 10 FPGA is an option – especially such as with very specific custom use cases, unique IP flows, data types, and/or multi-function workload flows. If you understand how to use FPGA's overall, similar to how they're used in other accelerated applications, then this may be a good acceleration solution.



# 深度学习误解

“深度学习离不开 GPU...”



## 错误

- 多数企业 (---) 将使用 CPU 满足他们的 AI 和深度学习需求
- 一些早期采用者 (---) 可能到了加速能力突破的关口<sup>1</sup>

It's a myth that you need to use GPUs for AI or deep learning.

As you see in this chart, most enterprises are below the blue line, successfully using Intel® Xeon® processors for AI and deep learning inference, and are now increasingly using them for deep learning training too, thanks to optimizations that have led to breakthrough performance increases. This performance continues to improve over time, and many enterprises will never need acceleration to meet their needs.

That said, at some point down the line in your AI journey, you may reach an inflection point where acceleration does become necessary. This could be driven by a particular usage model at initial deployment or once your application “takes off” with huge growth in inference demand (e.g. your app explodes like Instagram). However, for the initial proof-of-concept (POC) – which can take a few weeks to several months, don't waste your money on a limited-purpose GPU accelerator that will sit idle most of the time, and hardly save you any time (if at all) due to the added time required to manage/deploy/duplicate data/etc. If and when you are truly ready to benefit from acceleration, there will be exciting new options on the market to select from, some of which we cover in this presentation.

So, as a rule of thumb, if you are like most enterprises and are just beginning your AI journey, forget about acceleration and start with Xeon. It's already the standard for deep learning



inference in the data center, and is now more capable than ever for deep learning training thanks in large part to all the software optimizations in the past 1-2 years. At some point during the AI journey, you may need acceleration for a particular use case or because deep learning has grown to be significant in your overall compute mix, but cross that bridge when (and if) it comes... in fact, you may be more than satisfied with the continuous extension of Xeon AI performance that comes with each new generation, especially now that new AI features are being built into the silicon architecture.



The image features a solid blue background with a complex, abstract pattern of white, glowing particle tracks or energy lines. These tracks are composed of numerous small dots connected by thin lines, creating a sense of movement and depth. A central rectangular area is highlighted by two thin, horizontal white lines, within which the Chinese characters "软件" (Software) are displayed in a bold, white, sans-serif font.

软件



# 加速开发

借助开放式人工智能软件



With Intel AI, you can speed up development with open AI software.

Intel is investing in AI tools that get the most out of, and streamline development across, each hardware option in our portfolio. This ultimately accelerates total time-to-solution.

• **For application developers**—those who deploy solutions using AI-based algorithms—Intel has several tools to optimize performance and accelerate time-to-solution. For deep learning, the Intel Distribution of OpenVINO Toolkit facilitates model deployment for inference by converting and optimizing trained models for whichever hardware target is downstream. It offers support for models trained in TensorFlow, Caffe, and MXNet on CPU, integrated GPU, VPU (Movidius Myriad 2/Neural Compute Stick), and FPGA. Intel also launched the beta of a tool to help compress the end-to-end deep learning development cycle. This open source, scalable and extensible distributed deep learning platform, built on Kubernetes, is called Nauta (pronounced as ‘nau·ta’; means ‘sailor’ in Latin), formerly known as the Intel Deep Learning Studio.

• **For data scientists**—those who create AI-based algorithms—Intel contributes to and optimizes a set of open source libraries that are widely used for machine and deep learning. There are a number of such machine learning libraries that can be used to get the most out of Intel hardware today, spanning Python, R, and Distributed. For deep learning,



Intel aims to ensure that all the major deep learning frameworks and topologies run well on Intel hardware, and customers are of course free to choose whichever frameworks best suit their needs. We've been directly optimizing the most popular AI frameworks first, based on market demand, and producing huge improvements. Today, we have many optimized topologies available for TensorFlow, MXNet, Caffe2/PyTorch, and BigDL on Spark, and you can download and install the optimized version of these frameworks by clicking on the link in this slide. Going forward, we intend to enable even more frameworks like PaddlePaddle, CNTK & many more through the Intel nGraph compiler.

- **For library developers**—those who develop and optimize APIs, libraries, and frameworks to support new algorithms and topologies on the underlying hardware—Intel offers a host of foundational building blocks to get the most out of our hardware. Beginning on the left with the primitives category, the Data Analytics Acceleration Library and Intel Python distribution are important building blocks for machine learning. The DNN (deep neural network) open source libraries contain CPU-optimized functions that are most relevant for, you guessed it, deep learning model development. On the right side of this row is a description of the Intel nGraph library (formerly the Nervana Graph), which takes the computational graph from each deep learning framework and creates an intermediate representation, which is executed by calling the math accelerator software libraries of each Intel hardware target. This compiler reduces the need for framework and model direct optimization for each hardware target using low-level software and math accelerator libraries. Today, it supports Intel Xeon CPUs, GPU (CUDA), and the Crest family, with more hardware targets planned going forward.





社区



# 英特尔® 人工智能学院

面向开发人员、学生、讲师和初创公司

通过在线教程、网络研讨会、学生套件及支持论坛提高智能性

学习

开发

获取四周的英特尔® AI DevCloud 免费访问权限，使用您基于英特尔® 至强® 处理器的现有集群，或使用公有云服务

使用可用的教材、动手实验室等教授他人

讲授

分享

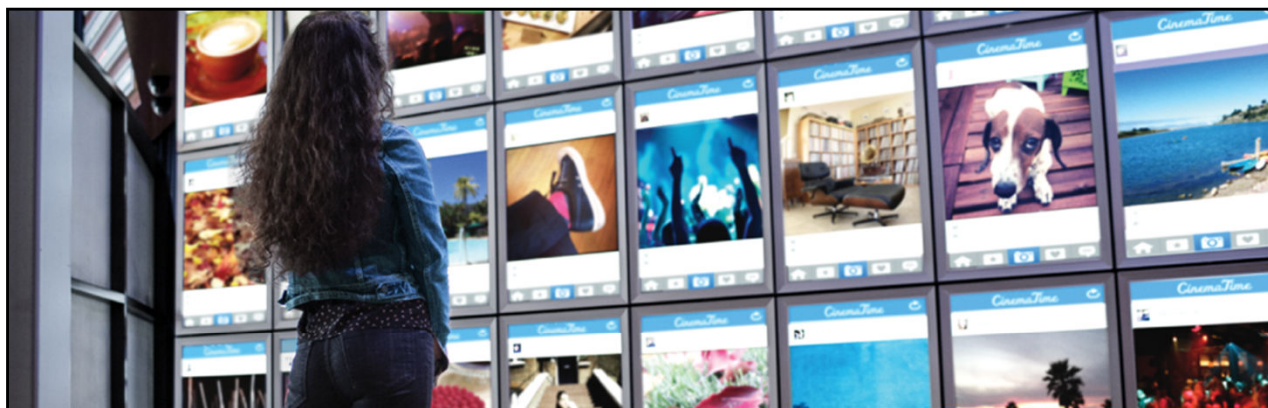
在行业与学术活动中及通过英特尔 AI 社区论坛在线展示您的创新成果

[software.intel.com/ai](https://software.intel.com/ai)

intel AI | 23

So where can you get started? The Intel AI academy is a great place to start for developers, students, instructors and startups. There, you can learn all about AI, download tools and resources to begin development with AI, find course materials to teach others & spread the knowledge, and share things that you've learned and created with the AI community. To get started, go to [software.intel.com/ai](https://software.intel.com/ai).





**SHARE** YOUR **STUDENT** PROJECT WITH THE WORLD

如欲了解更多信息，  
请访问 DevMesh

以英特尔® 校园大使身份分享您项目的机会

- 获得行业活动上的赞助演讲机会
- 大使实验室
- 学生研讨会
- 英特尔® Developer Mesh



# AI BUILDERS：生态系统

超过 100 家  
人工智能合作伙伴

## 跨垂直行业



## 垂直行业



## 水平




\* 其他的名称和图标可能随时间而更改

[Builders.intel.com/ai](https://builders.intel.com/ai)

intel AI | 25

And last but certainly not least, you can turn to Intel or one of our many ecosystem partners to help you get started on your AI journey. Visit [builders.intel.com/ai](https://builders.intel.com/ai) to find out more about one of our 100 and counting list of AI builder partners.





与英特尔  
携手合作,  
加速您的  
人工智能  
之旅

## 为何选择英特尔 AI？



### 简化人工智能

通过我们强大的社区



### 驾驭您的数据洪流

借助我们的数据层专家



### 选择任何合适的方法

从分析到深度学习



### 加速开发

借助开放的人工智能软件



### 随处部署人工智能

借助前所未有的可选硬件



### 自信扩展

在面向 IT 和云的引擎上

[www.intel.ai](http://www.intel.ai)



26

Intel is the only company that spans the entire AI journey. The Intel AI commitment to our customers is simple: we're here to help them break barriers between AI theory and reality. With Intel AI, our customers can **simplify AI, choose any approach, tame their data deluge, speed up development, deploy AI anywhere, and scale with confidence.** We look forward to building what was once thought to be impossible, with YOU. For more information about all these and more, visit [www.intel.ai](http://www.intel.ai)



# 资源

英特尔® 人工智能学院

<https://software.intel.com/en-us/ai>

英特尔® 人工智能学生套件

<https://software.intel.com/en-us/ai/courses>

英特尔® DevCloud

<https://software.intel.com/en-us/devcloud>

英特尔® 人工智能学院支持社区

<https://software.intel.com/en-us/forums/intel-optimized-ai-frameworks>

英特尔® DevMesh

<https://devmesh.intel.com>

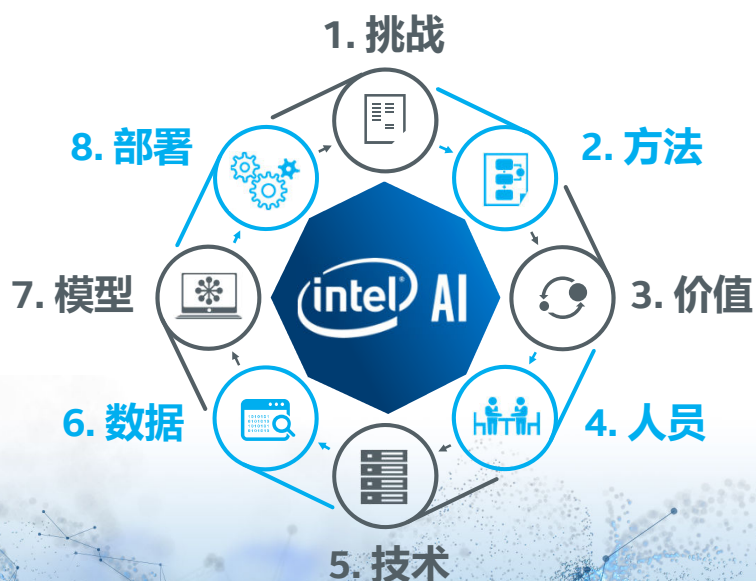




# 英特尔助力 人工智能之旅



# 人工智能之旅 介绍英特尔 案例研究



Before we venture any further, it's important to understand that implementing AI in your organization will be a journey, and to think about which technology partner can help you accelerate each step to take you full circle. **In the next four slides, we'll walk through an Intel AI case study to illustrate this journey.**





This is a case study based on Intel’ s AI engagement with an industrial sector company.

## 1.Challenge

This journey began with a survey of potential AI opportunities, starting with internal brainstorming, surveying the external landscape, and combing through the 70+ solutions in the Intel AI builders program. Once we identified some promising opportunities, the next step was to assess and rank the business value of implementing each AI solution.

## 2.Approach

The next step was to work with Intel’ s experts to identify the best approach (analytics, ML, DL, etc.) and estimate the associated complexity/cost of each solution. For example, building a new deep learning model from scratch is more costly than building off an existing deep learning model, which in turn is more costly than using a know machine learning method. We then plotted the top AI opportunities on a value/simplicity chart, it became clear which project would deliver the highest ROI: automating underwater industrial defect detection using deep learning image recognition.

## 3.Values

Before going any further down the chosen path, it’ s important to assess the “other”

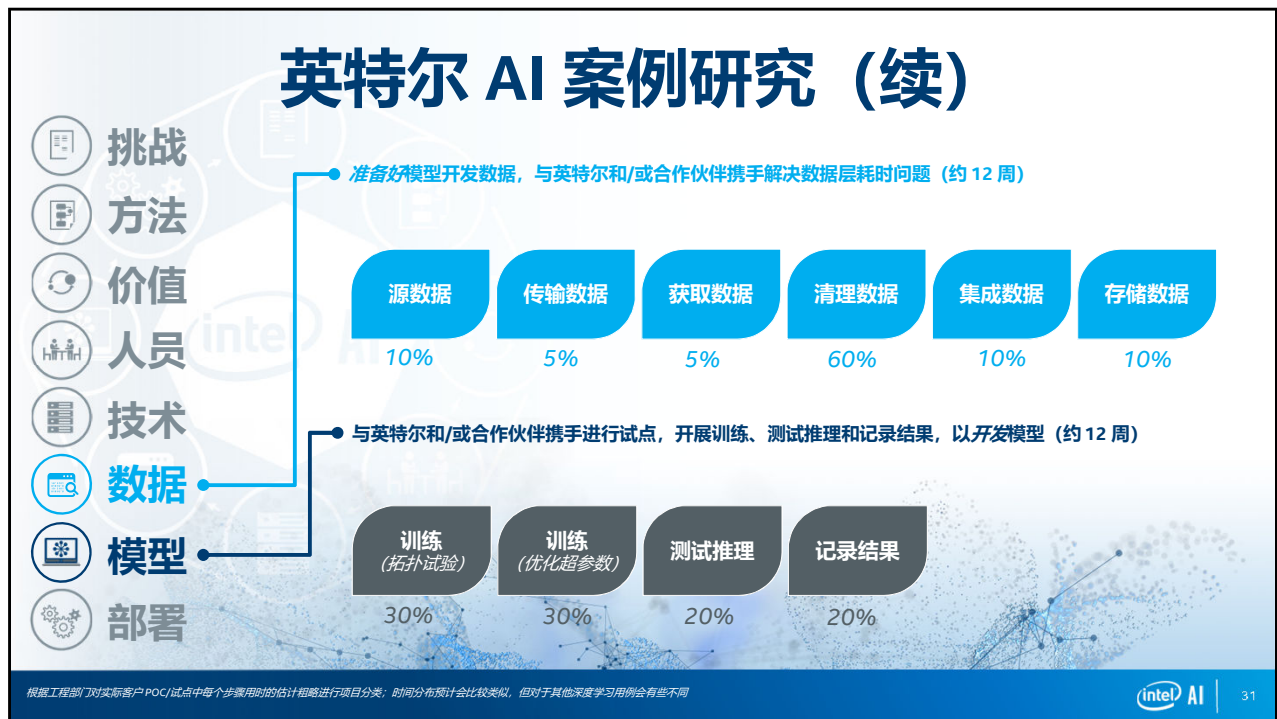


ramifications of an AI implementation, beyond the dollars and cents. In this case, we discussed the legal, social, and ethical issues that may arise, what we could do to mitigate them, and whether there we had any showstopper risks. **We also documented the assessment and mitigation plan to revisit if/when this pilot goes into production.**

#### **4. People**

The next step was to secure organizational buy-in and build up the right talent. This step is crucial, because if key stakeholders aren't ready to accept data-driven insights, then all the work ahead may be for naught. A classic example is the initial resistance to data analytics in sports, where general managers and scouts scoffed at the idea of computer algorithms outsmarting their years of experience and tribal knowledge. We used other Intel AI solution briefs and customer testimonials get buy-in, as well as ensure that the organization was ready to embrace the fact that AI development is *different*, involving more trial & error and uncertainty than traditional software development. **Next, we assessed the talent situation and determined that training up existing developers through Intel's free AI developer program was the best approach.**





While the time slice breakdowns you’ ll see on this slide are only based on this example, other projects will vary slightly but generally follow the same process.

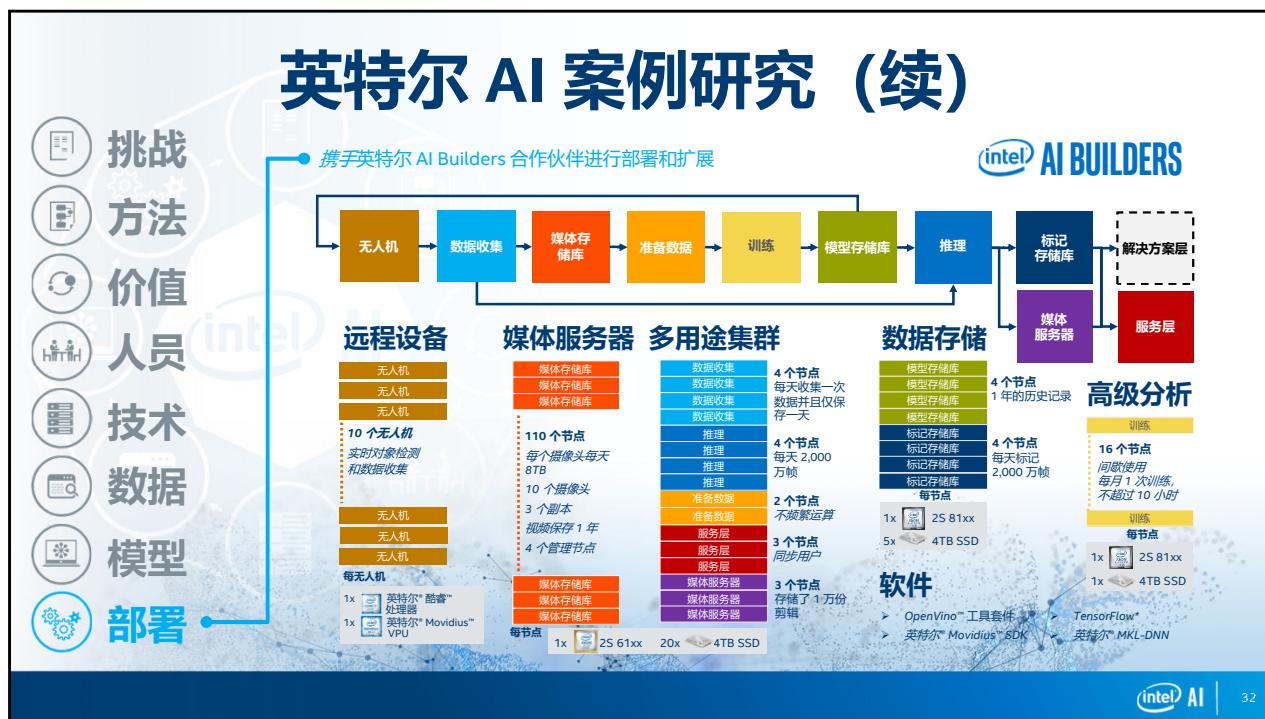
## 6.Data

One of the biggest barriers to AI, which is often overlooked, is getting your data ready. From sourcing to storing to preparing’ cleansing data for analysis, Intel worked with this customer to get their data layer right – a stage that took about as long as the actual model development itself!

## 7.Model

Once the data was ready, the team began experimenting with various topologies and tuning hyperparameters through iterative training runs. Once a sufficiently high accuracy was reached, the trained model was tested against a control data set, and inference results achieved a high enough accuracy to proceed to the cleanup & documentation phase. About 60% of the time was spent training, whereas the rest was testing & documentation.





## 8. Deploy

The final (and arguably most complex) stage is to take the pilot to production, deploying the model at scale.

In this case, our customer joined forces with a partner from the Intel AI Builders program to put together a “real world” AI solution.

The colorful block diagram at the top is a functional description of each step in this industrial defect detection scenario. There are 10 underwater drones that are equipped with video cameras to monitor heavy industrial equipment in order to detect potential defects. These drones capture videos of the underwater equipment, which is then ingested into the data center. Those videos are stored, for use in re-training the model and future reference, as well as passed to the inference cluster to determine if and where defects are. For re-training, human experts label images where the equipment was present or not, and where defects were present or not, in order to continue build the dataset and achieve even higher levels of accuracy. The latest trained models are stored, with one being deployed to perform object recognition inference on the drone (to aim the cameras at the equipment itself), and the other deployed to perform defect image recognition inference in the data center on the ingested video streams. As possible defects are identified, the inference output is sent to both the service layer (for



human audit) and the solution layer, where it is used as part of a larger decision process to determine whether to call a technician and/or shut down the equipment.

The stacks at the bottom of this slide illustrate the infrastructure – both hardware and software – underlying each colored step in the solution. This includes a whole lot of Xeon-based servers in the data center with SSD storage, Movidius VPU's in the drones, and Intel AI software like the OpenVINO toolkit, the Movidius SDK, and the latest Intel-optimized version of TensorFlow with MKL-DNN.

**THE BOTTOM LINE** here is that AI in the real world is much more involved than in the lab, and Intel & our partners are here to help you... not only with your deployment at scale, but to accelerate each and every step in your AI journey. Next, we'll see what Intel AI brings to the table.

----- BACKUP -----

The **functional layout (left)**:

- Feed in videos of the industrial equipment captured by a network of remote cameras on drones
- The streaming video input is a set of video clips that show the industrial equipment
- The output is a set of short video clips that pinpoint the specific areas of defects that need to be serviced
- All these color-coded boxes represent the various compute nodes in the data center to support this system
- Italic text is the capacity driver – what determines how much compute or storage capacity is needed at that stage
- In the training path, new videos for use in training come in and are stored in the media store, with capacity driven by the frequency of the import and the number of videos to be stored
- Human experts then review the videos, locate the defects of interest in the images and categorize their defectiveness
- These labeled images are then fed into a system used by the data scientist for developing and training the deep neural network (DNN)
- The output of their works – various experiments and trained models – are stored in the Model Store
- The current best models are moved to the Inference system where they will be used for real-time analysis of new never-before-seen videos coming from the cameras
- Now in the Inference path, as new videos come in to the Inference system, the DNN automatically identifies defects and stores this information in the Label Store
- The subset of interesting clips are stored in the Media Server.
- Whenever anyone wants to review the current videos of interest, they can login and issue requests to the Service Layer – could include a media player – that allows the user to see all the annotated videos that show defects.



### The **datacenter design (right)**:

- Before we explore the node count for each function, what drives that count?
  - Data ingest: frequency of ingest
  - 媒体存储库# videos stored
  - Feature engineering: frequency of new data
  - Training: model complexity, data set size and time-to-train requirement
  - Model store: frequency of training and model lifecycle requirements
  - Inference classification: frames/day
  - 标记存储库# labels & history
  - 媒体服务器# clips & history
  - Service Layer: simultaneous users
- 110 nodes for video storage – 10 cameras running all the time, keeping all the video for a year
- Going back to my earlier slide about the coming flood of data – this is what we’re talking about!
- And as we said before, we can’t just keep all that data forever. It’s too much. Needs to be processed and acted upon in near real time
- That’s what much of the rest of this infrastructure is for
- 4 nodes of Inference servers for running inference on all the video as it comes in
- Then we’ve got 4 nodes for Label storage and 3 nodes for Media Server to catch the output of the Inference process
- And 3 nodes of Service Layer to make those processed videos available to the human experts who can review them and take action
- The solution layer is where the inference output is used as an input to some larger analytics/decision process (e.g. calling a technician and shutting down the manufacturing line)
- So that’s a lot of hardware. And this is pretty typical layout for an end-to-end AI solution
- Thus, there are big opportunities here, once you dig into a real AI solution, you start to see how AI spending can grow at >50% CAGR for the next 8 years



# 主要内容

人工智能在现实中的用武之地不是实验室能比拟的

多数情况下，获取及准备数据与模型训练及分析阶段同样

整个过程一般耗时数周到数月才可完成



# 在课堂上探讨人工智能之旅

- 复杂的企业问题牵涉较广，不是课堂能讲明白的
- 选择较小的挑战，介绍解决问题的步骤，为您在未来解决企业问题提供借鉴
- 今天的人工智能之旅课程将聚焦：
  - 明确挑战
  - 技术选择
  - 获取数据集和探索性数据分析
  - 在 CPU、集成显卡和英特尔® Movidius™ 神经计算棒上训练和部署模型



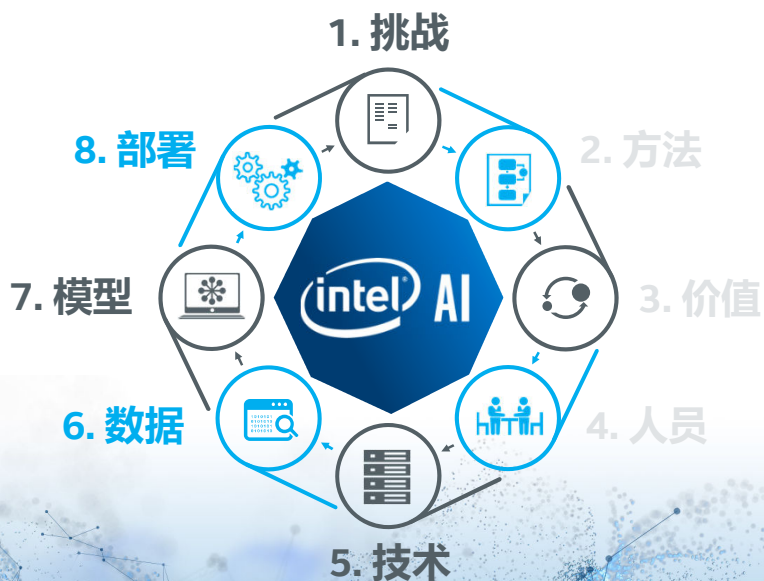




# 解决问题的实践



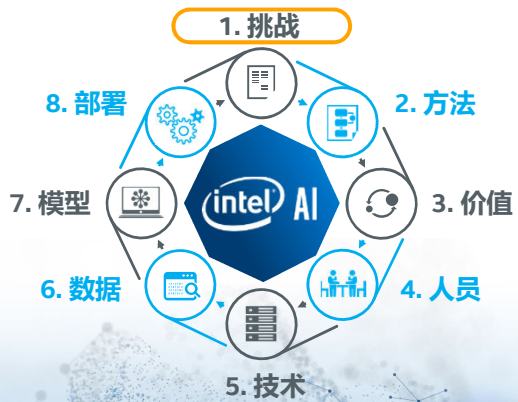
# 人工智能之旅 — 本课程将介绍的步骤



Before we venture any further, it's important to understand that implementing AI in your organization will be a journey, and to think about which technology partner can help you accelerate each step to take you full circle. **In the next four slides, we'll walk through an Intel AI case study to illustrate this journey.**



# 步骤 1 — 挑战



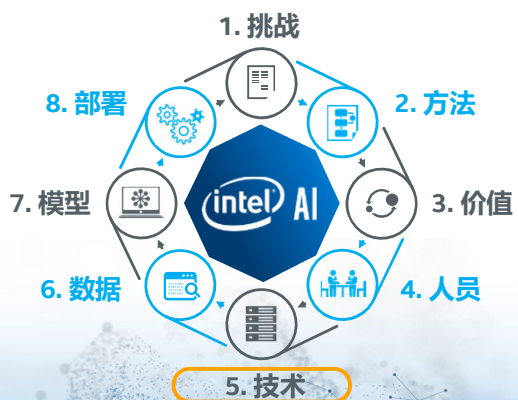
- 洞悉挑战 — 识别美国最常失窃的车辆
  - 图像识别问题
- 应用 — 交通监控
  - 扩展到车牌检测（不在本课范围内）



# 技术选择



## 步骤 5 — 训练和推理的计算选择



- 英特尔® AI DevCloud
- Amazon Web Services\* (AWS)
- Microsoft Azure\*
- Google 计算引擎\* (GCE)





英特尔® AI DEVCLOUD



# 英特尔® AI DEVCLOUD

- 英特尔® 人工智能学院成员可使用云托管硬件和软件平台进行学习、沙盒操作以及开始人工智能项目
- 英特尔® 至强® 可扩展处理器（英特尔® 至强® 金牌 6128 CPU @ 3.40GHz 24 核，2 路超线程，96 GB 平台 RAM（DDR4），200 GB 文件存储）
- **4 周初始访问，根据项目需求扩展**
- 英特尔® 人工智能学院支持社区提供技术支持
- 现已向所有人工智能学院成员开放
- <https://software.intel.com/en-us/devcloud>

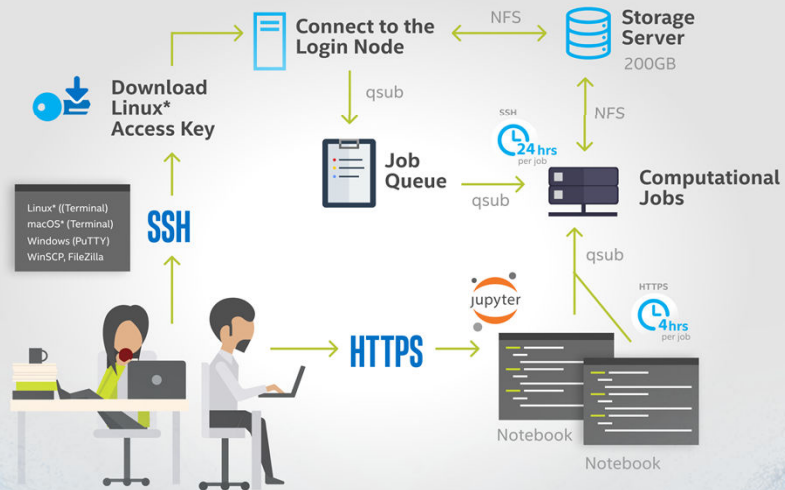


# 优化的软件 — 无需安装

- Python\* 2.7 和 3.6 英特尔® 分发版, 包括 NumPy、SciPy、pandas、scikit-learn、Jupyter、matplotlib 和 mpi4py
  - 英特尔® 优化的 Caffe\*
  - 英特尔优化 TensorFlow\*
  - 英特尔优化 Theano\*
  - Keras 库
  - 更多框架将被优化
- 英特尔® Parallel Studio XE 集群版及其附带的工具和库：
    - 英特尔 C、C++ 和 Fortran 编译器
    - 英特尔® MPI 库
    - 英特尔® OpenMP\* 库
    - 英特尔® 线程构建模块库
    - 英特尔® 数学核心函数库-DNN
    - 英特尔® 数据分析加速库



# DEVCLLOUD 概述







## 其他带英特尔 处理器支持的选择



# 选择您的云计算

## Amazon Web Services\* (AWS) :

- 名称 : C5 或 C5n
- vCPU : 2 - 72
- 内存 : 4gb - 144gb

## Microsoft Azure\* (Azure) :

- 名称 : Fsv2
- vCPU : 2 - 72
- 内存 : 4gb - 144gb

## Google 计算引擎\* (GCE) :

- 名称 : n1-highcpu
- vCPU : 2 - 96
- 内存 : 1.8gb - 86.4gb

## 您的计算选择 :

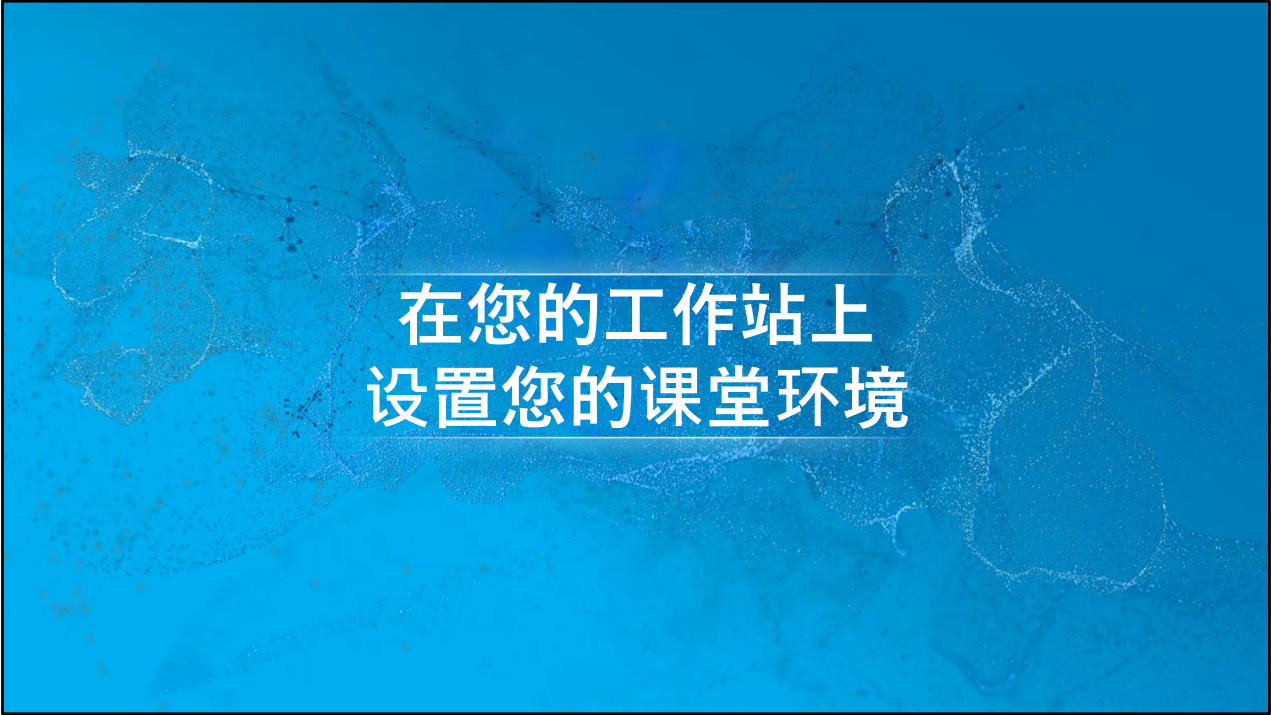
- 较好 : 英特尔® 至强® 可扩展处理器 (代号 Skylake) /最好 : 第二代英特尔® 至强® 可扩展处理器 (代号 Cascade Lake)
- AVX512 和 VNNI 支持
- 每个云服务提供商的计算密集型实例类型
- 内存和 vCPU 仅适用于您的数据集



45

Not all servers are equal, each CSP has different choices for servers. Depending on your favorite CSP we recommend looking for these types of instances to get the necessary Processors that support the optimized software features., e.g., AVX512 and or VNNI





在您的工作站上  
设置您的课堂环境



# 系统配置

## 支持的硬件：

- 第六代到第八代智能英特尔® 酷睿™ 处理器和英特尔® 至强® 处理器
- 采用英特尔® 核心显卡的英特尔奔腾® 处理器 N4200/5、N3350/5 或 N3450/5

## 支持的操作系统：

- Windows® 10 (64 位)
- Ubuntu® 16.04.3 LTS (64 位)
- CentOS® 7.4 (64 位)
- Yocto Project® 版本 Poky Jethro 2.0.3 (64 位)
- macOS® (64 位)

<https://software.intel.com/en-us/opencv-toolkit/hardware>





# 创建 ANACONDA 环境

1. 进入代码所在的根目录

2. 运行 `conda env create -f environment.yml` 创建环境。

3. 运行 `conda activate tf_training` 激活环境

4. `python -m ipykernel install --user --name tf_training --display-name "tf_training"`

5. 现在运行 `jupyter notebook` 启动 notebook

6. 在您的 notebook 中，选择 “Kernel -> Change kernel”，并选择 “tf\_training” 作为 kernel。

现在，您能够使用需要的所有库完成练习！

注：如果您在创建环境时遇到任何问题，请停用并删除环境，然后从步骤 1 开始操作。

```
conda deactivate 然后 conda env remove -n tf_training
```





## 在英特尔® AI DEV CLOUD 上设置您的课堂环境



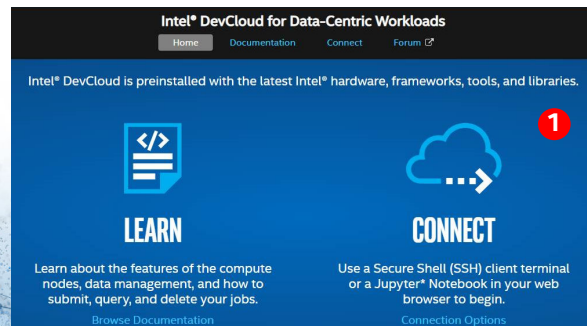
# 连接您的 DEVCLOUD 账户

注册英特尔® DevCloud 帐户

<https://software.intel.com/en-us/devcloud/datacenter>

点击 DevCloud 欢迎邮件中的 URL 以访问您的帐户。这会打开 DevCloud 主页。

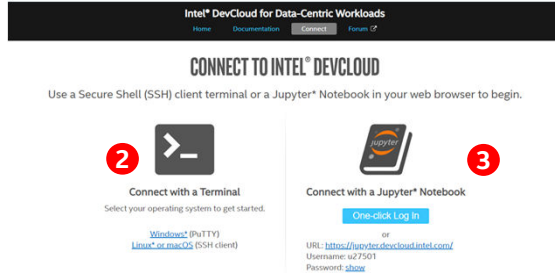
1. 点击 Connect 图标连接您的帐户。





# 连接您的 DEVCLLOUD 账户

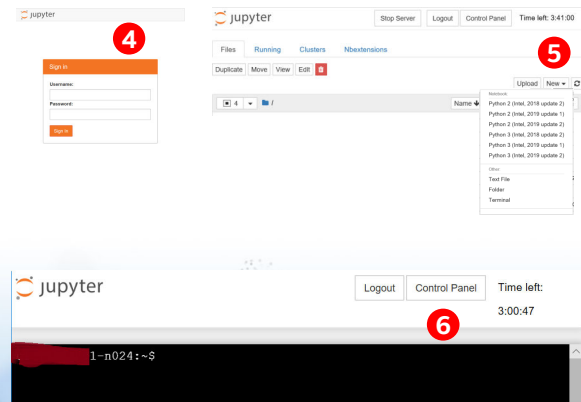
2. 从以下选项中选择一种方式连接。
  1. 通过终端连接 (Linux / Mac / Windows)
  2. 连接 Jupyter Notebook
3. 由于多数课堂练习将在 jupyter notebook 上完成，我们将选择选项 2。这会打开帮助我们获取用户名和密码的连接页面。
  - 退出此页面前复制用户名和密码。
  - 点击<https://jupyter.devcloud.intel.com/hub/login> 链接，导航至您的 jupyter notebook 帐户。





# 访问您的 DEVCLOUD JUPYTER NOTEBOOK 帐户

4. 输入之前复制的用户名和密码，访问您的 jupyter notebook 帐户。
5. 点击页面右侧的“New”菜单，并选择“Terminal”以访问终端。
6. 现在，您已通过 jupyter notebook 连接到您的 DevCloud 帐户。您可随时点击“Control Panel”按钮返回 jupyter 主页。



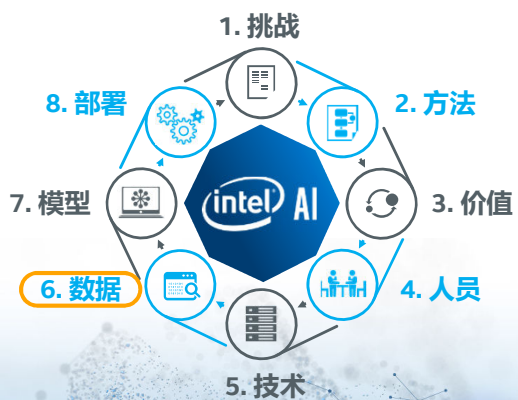




## 探索性数据分析



## 步骤 6 — 探索性数据分析



- 获取启动数据集
- 数据的初始评估
- 准备用于解决当下问题的数据集
  - 识别相关类别和图像
  - 预处理
  - 数据增强



# 获取启动数据集

- 查找与特定问题有关或匹配的现有数据集
  - 节省时间和资金
  - 利用其他人的工作成果
  - 利用积累的知识开展未来的项目
  - 我们开始使用 VMMRdb 数据集 (<http://vmrdb.cecsresearch.org/>)

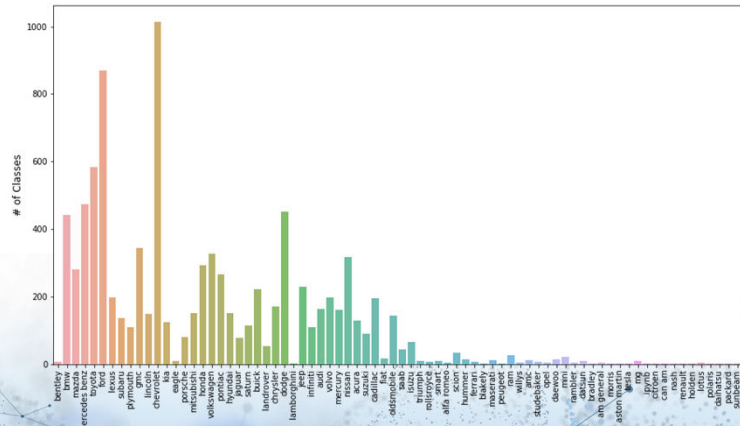


# 数据集的初始评估

汽车制造和型号识别数据集 (VMMRdb) :

- <http://vmmrdb.cecsresearch.org/>
- 大规模和多样性
- 从 Craigslist 中收集图像
- 包含 9170 个类别
- 识别出 76 家汽车制造商
- 共 291,752 幅图像
- 生产时间介于 1950 年到 2016 年
- 探索下面的 VMMR 数据集

Optional-Explore-VMMR.ipynb



汽车制造商



# 帮助解决汽车失窃挑战的数据集

**Hottest Wheels: 美国最常失窃的新车和二手车** (<https://www.forbes.com/sites/jimgorzelay/2018/09/18/hottest-wheels-the-most-stolen-new-and-used-cars-in-the-u-s/#3e9577545258>)

选择经常失窃的 10 个车型 — 缩短训练时间

- 本田思域 (1998) : 45,062
- 本田雅阁 (1997) : 43,764
- 福特 F-150 (2006) : 35,105
- 雪佛兰西尔维拉多 (2004) : 30,056 # 指出 2017 年每个车型的失窃数量
- 丰田佳美 (2017) : 17,276
- 日产 Altima (2016) : 13,358
- 丰田花冠 (2016) : 12,337
- 道奇/Ram Pickup (2001) : 12,004
- GMC Sierra (2017) : 10,865
- 雪佛兰黑斑羚 (2008) : 9,487

The problem we are trying to solve is based on the hottest wheels – most stolen cars.



# 准备解决挑战所需的数据集

- 确定不同年份车辆的失窃情况（基于外观相似性）

- 提供更多示例

- |                              |                                |
|------------------------------|--------------------------------|
| - 本田思域（1998）：45,062          | → 本田思域（1997 - 1998）            |
| - 本田雅阁（1997）：43,764          | → 本田雅阁（1996 - 1997）            |
| - 福特 F-150（2006）：35,105      | → 福特 F150（2005 - 2007）         |
| - 雪佛兰西尔维拉多（2004）：30,056      | → 雪佛兰西尔维拉多（2003 - 2004）        |
| - 丰田佳美（2017）：17,276          | → 丰田佳美（2012 - 2014）            |
| - 日产 Altima（2016）：13,358     | → 日产 Altima（2013 - 2015）       |
| - 丰田花冠（2016）：12,337          | → 丰田花冠（2011 - 2013）            |
| - 道奇/Ram Pickup（2001）：12,004 | → 道奇 Ram 1500（1995 - 2001）     |
| - GMC Sierra（2017）：10,865    | → GMC Sierra 1500（2007 - 2013） |
| - 雪佛兰黑斑羚（2008）：9,487         | → 雪佛兰黑斑羚（2007 - 2009）          |



# 对数据集进行预处理

- 获取并目视检测数据集
- 图像预处理
  - 解决数据集不平衡问题
  - 将数据集用于训练、验证和测试组
  - 增强训练数据
    - 限制训练和测试数据之间的重叠
    - 充足的测试和验证数据集
- 完成笔记本：第 1 部分 — [Exploratory\\_Data\\_Analysis.ipynb](#)



# 检查数据集

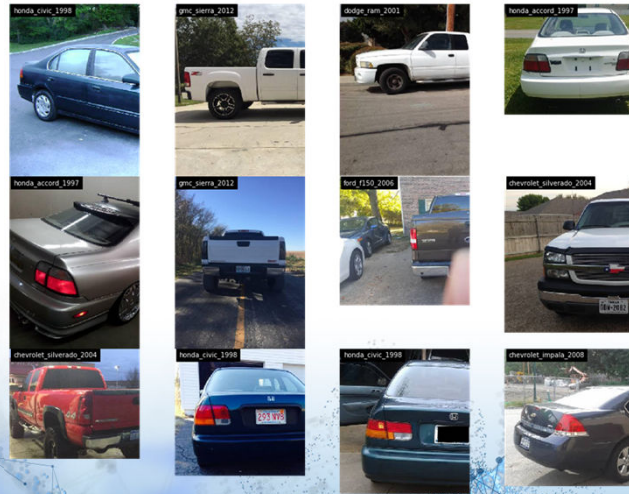
- 目视检查数据集

- 注意不同之处

- › ¾ 视图
    - › 前视图
    - › 后视图
    - › 侧视图等
    - › 图像高宽比不同

- 示例类别名称:

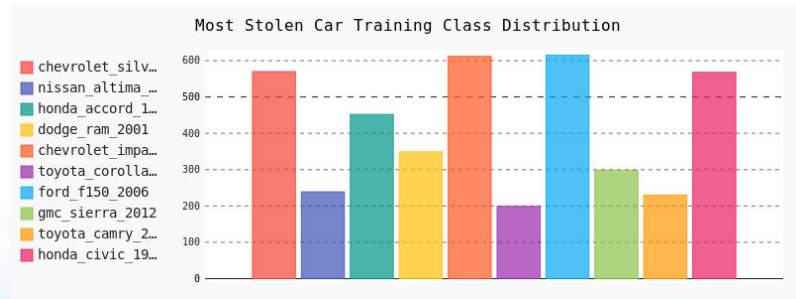
- 制造商
  - 车型
  - 年份





# 数据创建

- 本田思域 (1998)
- 本田雅阁 (1997)
- 福特 F-150 (2006)
- 雪佛兰西尔维拉多 (2004)
- 丰田佳美 (2014)
- 日产 Altima (2014)
- 丰田花冠 (2013)
- 道奇/Ram Pickup (2001)
- GMC Sierra (2012)
- 雪佛兰黑斑羚 (2008)



We wanted a category of everything but the top 10 most stolen. After experimentation we discovered that it had too many similarities to the other 10 categories and we ended retraining without the others category and we got significant improvement in prediction.



# 预处理和增强

## 预处理

- 删除原始数据中不一致和不完整的部分，在清理后将数据用于模型
- 技术：
  - 黑色背景
  - 尺度调整，灰度调整
  - 样本均值化，归一化
  - 特征均值化，归一化
  - RGB → BGR

## 数据增强

- 改善数据集的数量与质量
- 可用于数据集较小或一些类别的数据比其他类别较少的情况
- 技术：
  - 旋转
  - 水平和垂直移动，翻转
  - 缩放和修剪

详细了解 `Optional-VMR_ImageProcessing_DataAugmentation.ipynb` 中的预处理和增强方法



# 预处理



灰度调整



样本均值化



样本归一化

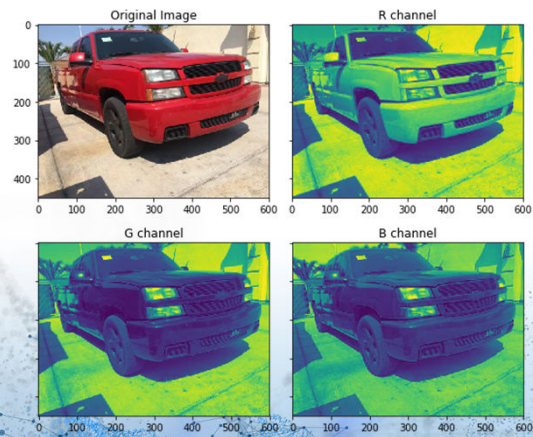


旋转



# RGB 通道

- 图像由像素组成
- 像素由红色、绿色、蓝色、通道等组成。





## RGB - BGR

- 取决于所需的网络选择 RGB-BGR 转换。
- 处理该任务的一个方式是使用 Keras\* `preprocess_input`

>> `keras.preprocessing.image.ImageDataGenerator(preprocessing_function=preprocess_input)`





# 数据增强

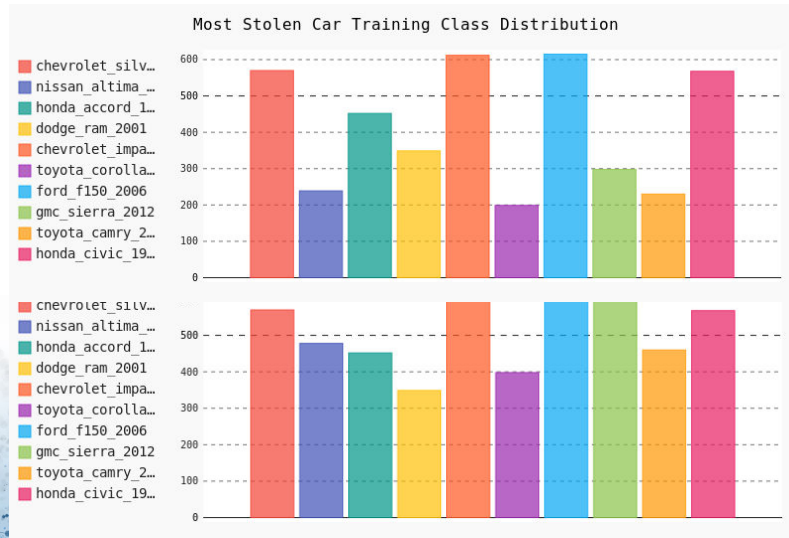
- 对训练集中的数据量较少的类别进行过采样





# 总结

预处理前



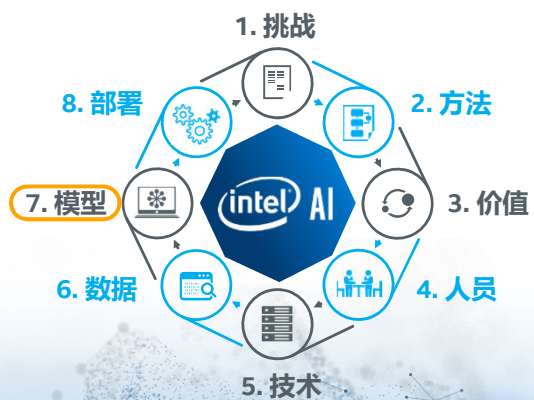
预处理后



# 训练阶段



## 步骤 7 — 训练/模型阶段



### • 生成训练后模型需要多个步骤

- 选择框架 (Tensorflow\*、Caffe\*、PyTorch)
- 选择网络 (InceptionV3、VGG16、MobileNet、ResNet 等或定制网络)
- 模型训练及调优，以提高性能
- 超参数调优
- 生成训练后模型 (冻结图形/caffemodel 等)



# 选择框架



# 选择框架的决策考量

英特尔在优化  
哪些框架？

选择特定框架的决策  
因素有哪些？

我们为何选择  
Tensorflow？



# 优化的深度学习框架

安装英特尔优化的框架和特定拓扑

## 英特尔优化的框架



更多框架正在优化:  等。

立即开始: <https://software.intel.com/en-us/frameworks>

另请参阅: 面向 Python (Scikit-learn, Pandas, NumPy) 、 R (Cart, randomForest, e1071) 和 Distributed (MLlib on Spark, Mahout) 的机器学习函数库  
\* 目前可用性有限  
\* 其他的名称和图标可能是其他所有者的资产。



How do you unleash all that deep learning performance on the Intel® Xeon® Processor? Well, you need to install an Intel-optimized framework to get started.

Intel aims to ensure that all major DL frameworks and topologies will run well on Intel Architecture, and customers are free to choose whichever framework(s) best suit their needs. We've been directly optimizing the most popular AI frameworks for Intel Architecture (based on market demand) and producing huge speedups. We intend to enable even more frameworks in the future through the Intel® nGraph™ Compiler. Please note that each of these frameworks have a varying degree of optimization and configuration protocols, so visit [ai.intel.com/framework-optimizations/](https://ai.intel.com/framework-optimizations/) for full details. Of special note is the BigDL framework that's been getting a LOT of traction lately with customers who want an easy way to achieve high-performance deep learning on their existing big data/analytics infrastructure. BigDL is a distributed deep learning library for Spark that can run directly on top of existing Spark or Apache Hadoop\* clusters with support for Scala or Python programming languages.



#### REACTIVE ONLY:

At the bottom-right, you can see the badge for the neon framework, which Intel also develops and maintains. Neon is our innovation framework, which means we use it to implement new DL research quickly by adding support for new DL layer types, etc. Neon is also our reference framework, which means we will typically conduct new performance benchmarking first on Neon. Customers are free to choose whatever framework they want. We are not promoting Neon over other frameworks or trying to get customers to switch from other frameworks to Neon.



# CAFFE / TENSORFLOW / PYTORCH 框架

利用机器学习框架/库可以更快地开发深度神经网络模型。许多框架可供选择，选好很重要。选择时要考虑的一些因素包括：

1. 开源和采纳程度
2. CPU 优化
3. 图形可视化
4. 调试
5. 库管理
6. 推理目标（CPU/集成显卡/英特尔® Movidius™ 神经计算棒/FPGA）

在考虑所有这些因素后，我们决定使用 Google 深度学习框架 **TensorFlow**



# 我们为何选择 TENSORFLOW?

框架选择基于以下要点:

## 开源和高采纳程度

- 支持更多特性, 还具有 “contrib” 包, 可用于创建更多模型, 从而支持更多高级功能。

## CPU 优化

- 带有 CPU 优化的 TensorFlow 可将训练速度提升多达 14 倍, 将推理速度提升多达 3.2 倍。TensorFlow 非常灵活, 可以支持对新的深度学习模型/拓扑和系统级优化进行试验。英特尔优化已经部署到上游, 并且是公共 TensorFlow\* Github 存储库的一部分。

## 推理目标 (CPU/GPU/Movidius/FPGA)

- TensorFlow 可以在不同类型的设备上扩展或部署, 包括 CPU 和 GPU, 并在手机等小型设备上推理。TensorFlow 与 CPU、GPU、TPU无缝集成, 无需任何直接配置。支持小型、移动、以及用于在服务器端部署的 TF serving。TensorFlow 图是可导出的图形 — pb/onnx



# 我们为何选择 TENSORFLOW?

框架选择基于以下要点:

- **图形可视化:** 相比于 Torch 和 Theano 等直接竞争对手, TensorFlow 具有更出色的计算图形可视化性能 (得益于 Tensor Board)。
- **调试:** TensorFlow 使用其调试器 “tfdbg” TensorFlow Debugging, 该调试器支持您执行图形的子部分, 以观察运行图的状态。
- **库管理:** TensorFlow 具有多种优势, 包括一致的性能、快速的更新和定期发布带有新特性的新版本。本课程使用 Keras, 其支持更轻松地迁移至 TensorFlow 2.0 以支持模型训练和测试。





# 选择网络



# 如何选择网络？

我们在开始实施这个项目时，便考虑到计划在作为最终部署平台的边缘设备上推理。为此，在选择拓扑或网络时，我们总是考虑三大要素：训练时间、规模和推理速度。

- **训练时间：**根据所需的层数和计算，网络训练时间可能有长有短。计算时间和程序员时间是昂贵的资源，所以我们希望减少训练时间。
- **规模：**由于我们的目标是边缘设备和英特尔® Movidius™ 神经计算棒，因此我们必须考虑内存中允许的网络大小以及支持的网络。
- **推理速度：**通常网络越深越大，推理速度越慢。在我们的用例中，我们使用的是实时视频流；我们需要至少每秒 10 帧的推理。
- **准确度：**拥有一个准确的模型同样重要。尽管大多数预训练模型都发布了其准确度数据，但我们仍然需要了解它们在数据集上的表现。



# INCEPTION V3 - VGG16 - MOBILENET 网络

我们决定在边缘设备（CPU、集成 GPU、英特尔® Movidius™ 神经计算棒）当前支持的三个网络上训练数据集。

原创论文\* 是在 ResNet-50 上进行训练的。但是，英特尔® Movidius™ 神经计算棒目前不支持它。

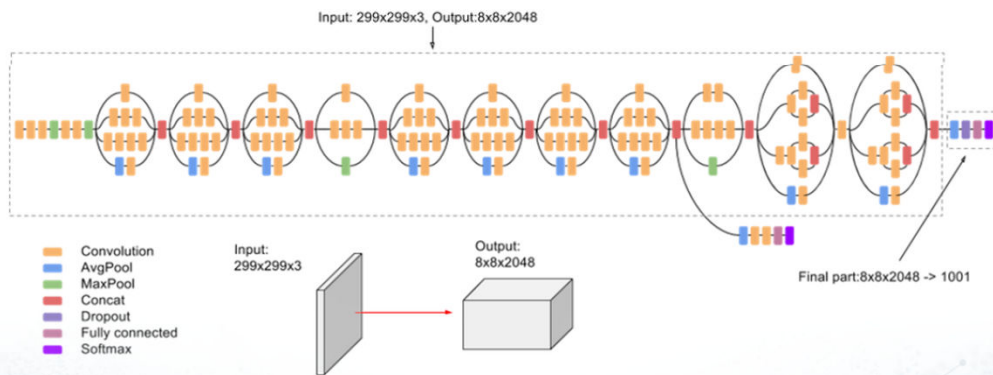
我们用于训练模型的支持网络包括：

- Inception v3
- VGG16
- MobileNet

[\\*http://vmrdb.cecsresearch.org/papers/VMMR\\_TSWC.pdf](http://vmrdb.cecsresearch.org/papers/VMMR_TSWC.pdf)



# INCEPTION V3



<https://arxiv.org/abs/1512.00567>

ImageNet 2015

Szegedy, et al.2014

Idea: network would want to use different receptive fields

Want computational efficiency

Also want to have sparse activations of groups of neurons

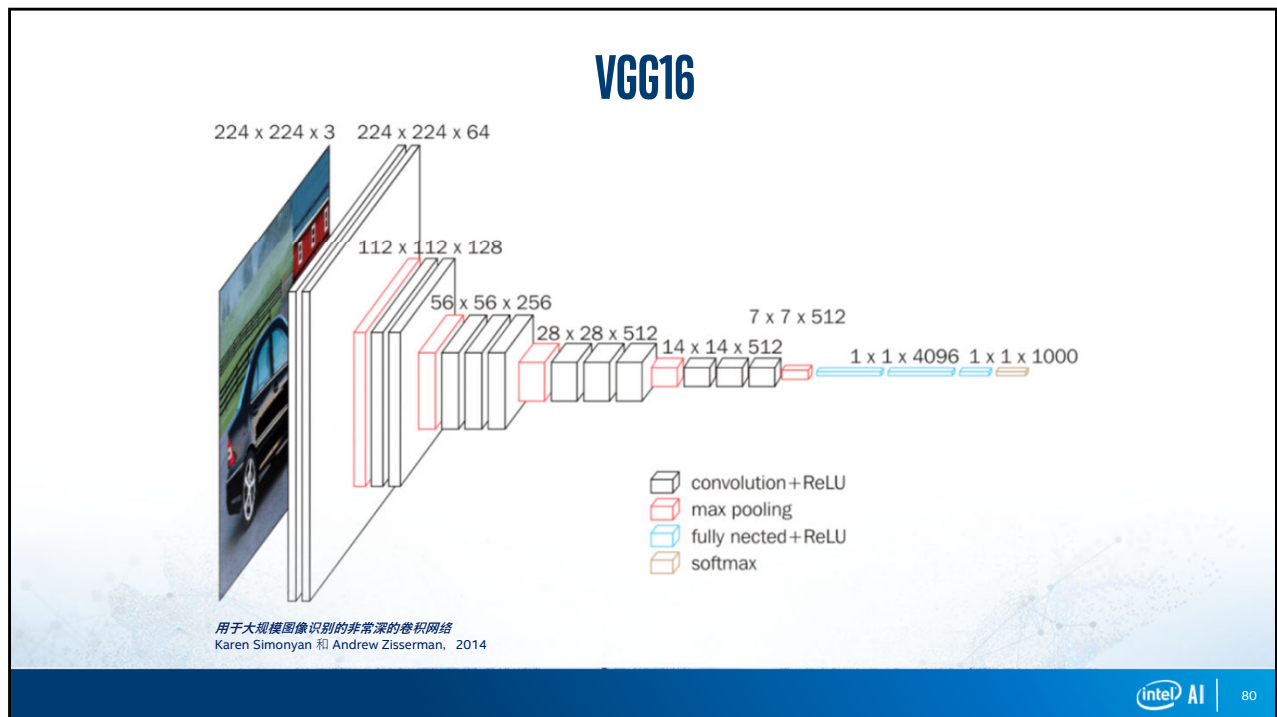
Hebbian principle: “Fire together, wire together”

Solution: Turn each layer into branches of convolutions

Each branch handles smaller portion of workload

Concatenate different branches at the end





One of the first architectures to experiment with many layers (more is better approach)

Uses multiple 3x3 convolutions to simulate larger kernels with fewer parameters

two 3x3 convolutions are equal to one 5x5

three 3x3 convolutions are equal to one 7x7

3x3xcxc - 9c2

7x7xcxc - 49c2



# MOBILENET

Table 1. MobileNet Body Architecture

Type / Stride	Filter Shape	Input Size
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv dw / s1	$3 \times 3 \times 32 \text{ dw}$	$112 \times 112 \times 32$
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Conv dw / s2	$3 \times 3 \times 64 \text{ dw}$	$112 \times 112 \times 64$
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Conv dw / s1	$3 \times 3 \times 128 \text{ dw}$	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Conv dw / s2	$3 \times 3 \times 128 \text{ dw}$	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv dw / s1	$3 \times 3 \times 256 \text{ dw}$	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Conv dw / s2	$3 \times 3 \times 256 \text{ dw}$	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
5x Conv dw / s1	$3 \times 3 \times 512 \text{ dw}$	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 512$	$14 \times 14 \times 512$
Conv dw / s2	$3 \times 3 \times 512 \text{ dw}$	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Conv dw / s2	$3 \times 3 \times 1024 \text{ dw}$	$7 \times 7 \times 1024$
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Avg Pool / s1	Pool $7 \times 7$	$7 \times 7 \times 1024$
FC / s1	$1024 \times 1000$	$1 \times 1 \times 1024$
Softmax / s1	Classifier	$1 \times 1 \times 1000$

<https://arxiv.org/pdf/1704.04861.pdf>

Picked initially due to it' s small nature

Uses global hyperparameters that efficiently tradeoff between latency and accuracy

These hyper-parameters allow the model builder to choose the right sized model for their application based on the constraints of the problem



## INCEPTION V3 - VGG16 - MOBILENET

在根据之前讨论的条件训练和比较结果后，我们最终选择的网络是 **Inception V3**。

我们比较了三个网络：

- Mobilenet 是最不精确的模型（74%），但最小（16mb）
- VGG16 是最精确的（89%），但最大（528mb）
- InceptionV3 的准确度居中（83%），大小也居中（92mb）

As you will see in the hands on section your results will be similar



# 总结

根据您的项目需求，框架和拓扑的选择将有所不同。

- 训练时间
- 模型大小
- 推理速度
- 可接受的准确度

对于这些选择，没有一刀切的方法，只有不断试错才可找到最佳解决方案。

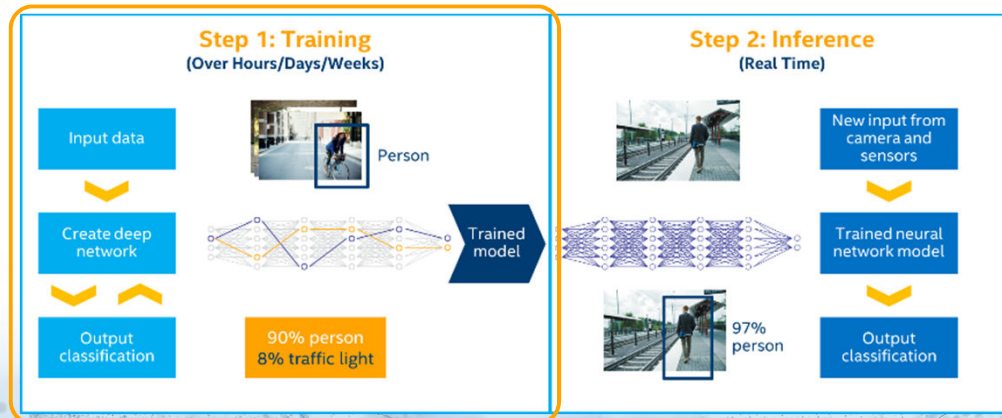




# JUPYTER NOTEBOOK 练习



# 训练和推理工作流程



完成笔记本：第 2 部分 - Training\_InceptionV3.ipynb

1. Inference on the PC is the process of performing computations on custom or specialized trained AI models in systems where limitations for size, power, and real-time performance are required to ensure success.



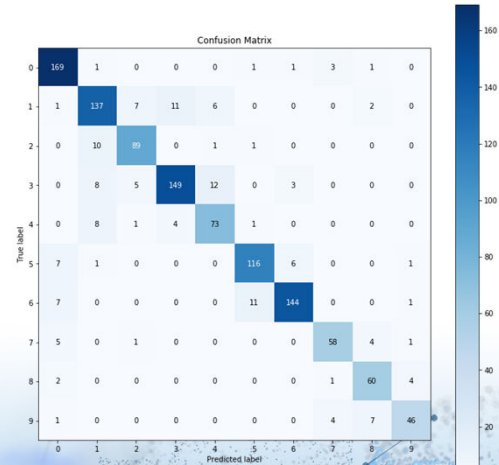
## (可选) 使用 VGG16 和 MOBILENET 训练

- 试用 [Optional-Training\\_VGG16.ipynb](#)
- 试用 [Optional-Training\\_Mobilenet.ipynb](#)
- 看看您的训练结果与 inceptionV3 有何不同



# 模型分析

- 了解如何通过使用不同的指标和图表分析我们的模型，以解释训练结果
  - 混淆矩阵
  - 分类报告
  - 精确率召回率图
  - ROC 图
- 完成笔记本 — 第 3 部分 - `Model_Analysis.ipynb`

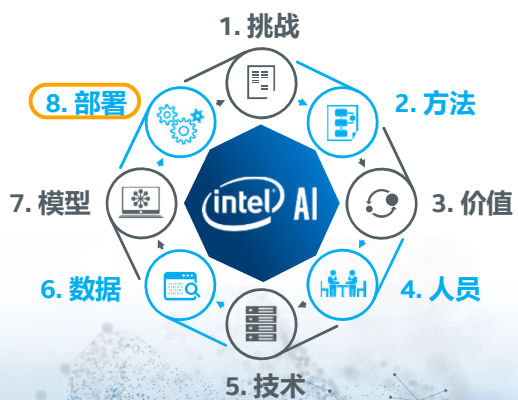




## 部署阶段



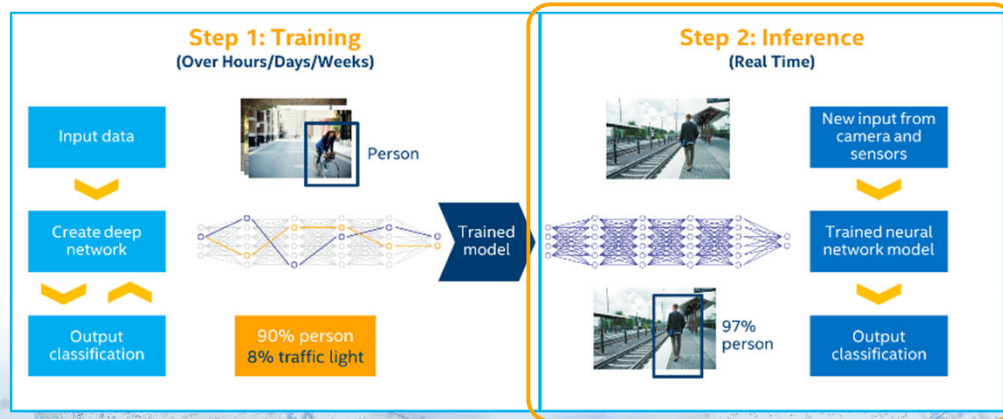
## 步骤 8 — 部署阶段



- 部署或推理是什么意思？
- 部署到边缘是什么意思？
- 了解英特尔® OpenVINO™ 工具套件分发版
  - 了解如何部署到 CPU、集成显卡、英特尔® Movidius™ 神经计算棒



# 部署/推理是什么意思？



1. Inference on the PC is the process of performing computations on custom or specialized trained AI models in systems where limitations for size, power, and real-time performance are required to ensure success.



# 边缘推理是什么？

模型的实时评估受到电源、延迟和内存方面的限制

需要专门针对上述限制调整的人工智能模型

例如，SqueezeNet 等模型针对在 PC 和嵌入式设备上图像推理进行了调整

- <https://towardsdatascience.com/deep-learning-on-the-edge-9181693f466c>
- 1.Bandwidth and Latency
- 2.Security and Decentralization
- 3.Job Specific Usage (Customization)





# 使用英特尔® OPENVINO™ 工具套件分发版进行边缘推理





# 用例



# 人数统计解决方案

(安装英特尔® OPENVINO™ 工具套件分发版时提供)

**描述** 一种应用，能够计算特定输入视频帧中的人数、迄今检测到的累积人数以及一个人出现在屏幕上的持续时间。此解决方案可用于零售店中的人员流量监视器。店主可以利用这些数据来优化人员配置、分析商店的不同区域并确定客流最大的时段等。该应用使用“ResMobNet\_v4 (LReLU) with single SSD head”模型作为其主干网。

**用例** 商店监控、视频监控、人流监控等

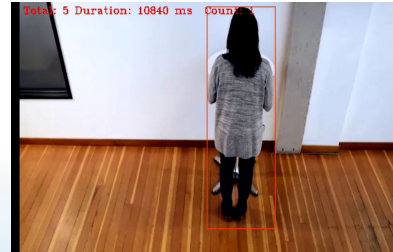
**软件要求** OpenVINO

**硬件要求** 英特尔酷睿系统、英特尔集成 GPU、Movidius VPU

**输入源** 本地存储的视频

**应用代码库** C++ API

**用户界面** 离线视频流





# 微情绪识别解决方案

(安装英特尔® OPENVINO™ 工具套件分发版时提供)

**描述** 此应用演示如何使用英特尔® 硬件和软件工具创建微情绪识别解决方案。此解决方案将情绪分为五类：“中性”、“快乐”、“悲伤”、“惊喜”、“愤怒”。它可用于市场研究行业的行为分析解决方案，帮助捕捉和分析客户产品交互视频源，进而优化营销策略。该应用使用包含两个模型的管道，一个使用基于深度卷积的默认 **MobileNet** 主干网，另一个为全卷积网络。

**用例** 情绪识别可用于采访、市场调研、视频监控等

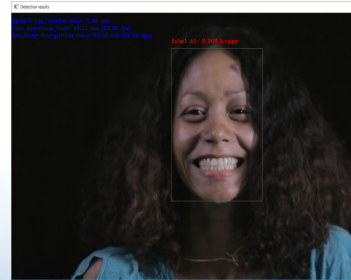
**软件要求** OpenVINO

**硬件要求** 英特尔酷睿系统、英特尔集成 GPU、Movidius VPU

**输入源** 本地存储的视频

**应用代码库** C++ API

**用户界面** 离线视频流



<https://towardsdatascience.com/background-removal-with-deep-learning-c4f2104b3157>



The background of the slide is a solid blue color with a complex, abstract pattern of white, glowing particle tracks or energy lines. These tracks are scattered across the blue field, some appearing as thin, straight lines while others form more intricate, branching or swirling shapes. The overall effect is reminiscent of a particle detector or a microscopic view of energy interactions.

## 预训练模型和示例



# 针对英特尔架构优化的预训练模型

OpenVINO™ 工具套件包括经过优化的预训练模型，能够在英特尔® 处理器上加快开发速度并改进深度学习推理。使用这些模型进行开发和生产部署，无需搜索或训练自己的模型。

## 预训练模型

- 年龄和性别
- 面部检测 – 标准和增强
- 头部位置
- 人员检测 – 视平线和高角度检测
- 检测人员、车辆和自行车
- 车牌检测：小巧、前置
- 车辆元数据
- 车辆检测
- 零售环境
- 行人检测
- 行人和车辆检测
- 交叉口人员特征识别
- 情绪识别
- 识别不同视频中的人员 – 标准和增强
- 识别路边对象
- 高级路边识别
- 人员检测和行为识别
- 人员再识别 – 超小/超快
- 面部再识别
- 特征回归



# 借助深度学习示例和计算机视觉算法节省时间

## 示例

这些示例代码中，对公共模型及英特尔预训练模型使用模型优化器和推理引擎。

- 对象检测
- 标准和管道图像识别
- 安全障碍
- 使用 Asynch API，将对象检测用于 Single Shot Multibox Detector (SSD)
- 对象检测 SSD
- 神经风格转换
- Hello 推理分类
- 交互式面部检测
- 图像分割
- 验证应用
- 多渠道面部检测

## 计算机视觉算法

使用预训练模型，通过高度优化、随时可部署、自定义构建的算法快速开始实施您的视觉应用。

- 面部检测器
- 年龄和性别识别器
- 摄像头篡改检测器
- 情绪识别器
- 人员再识别
- 交叉口对象检测器

Sharpen the difference





# 英特尔® 开放视觉接口和神经网络 优化（OPENVINO™）工具套件



# OPENVINO™ 工具套件的优势

充分利用英特尔® 处理器的强大功能：CPU、带集成显卡的 CPU、FPGA、VPU

## 提升性能

使用英特尔计算机视觉加速器。提高代码性能。支持异构处理和异步执行。

## 整合深度学习

使用通用 API 和 10 个预训练模型，发掘基于卷积神经网络（CNN）的深度学习推理潜力。

提升高达  
10 倍\*

## 加速开发

使用优化的 OpenCV\* 和 OpenVX\* 功能库和超过 15 个示例缩短时间。仅需开发一次，即可部署到当前和未来的英特尔架构设备。

## 创新和定制

使用 OpenCV\* 中不断扩展的 OpenCL™ 基础库添加您自己的独有代码。

\*与英特尔® 深度学习部署工具套件中的某些标准框架模型和英特尔优化模型相比，性能提升 10 倍。参见性能指标评测页。  
基准性能测试结果在实施近期针对 Spectre 和 Meltdown 漏洞的软件补丁和固件更新之前发布。实施更新后，这些结果可能不再适用于您的设备或系统。有关性能和基准测试结果的更完整信息，请访问：  
<https://www.intel.cn/content/www/cn/zh/benchmarks/benchmark.html>

\*\*某些技术规格和特定处理器/SKU 适用。如欲了解更多详细信息，请参见 [白皮书](#)。  
OpenVX 和 OpenVX 标识是 Khronos Group 的商标。  
OpenCL 和 OpenCL 标识是苹果公司的商标，需获得 Khronos 的许可方能使用



- Using a common API across CPU, GPU, FPGA and Movidius VPU allows for heterogeneity, fallback to CPU and/or GPU and no need to recode when testing on different HW.
- 10x improvement was achieved by GE when comparing a standard TensorFlow model they were using for image classification vs. when they used the DLDT. This was achieved on a Xeon system and allowed GE to increase their performance without the need of adding discrete GPU cards.
- Over, 40 models and samples are included in the package. In addition, there is a model downloader that will also download public models.
- Easy way to create an operation that isn't covered by the IE out of the box. You can express that as a composition of existing IE ops or register the op in the MO and connect it to the entirely new IE layer in C++ or OpenCL. The existed layers have been reorganized to “core” (general primitives) and “extensions” (topology-specific, for example Detection Output for SSD). These “extensions” should be built and loaded explicitly.



- Speed development - auto fallback to CPU/GPU for FPGA



# 深度学习对比传统计算机视觉

OpenVINO™ 具有用于端到端视觉管道的工具



## Key Takeaways

- OpenVINO™ has tools for both Traditional and Deep Learning CV
- Multiple Intel tools (Media SDK, OpenVINO™, ISS) work together to provide a complete CV pipeline optimization solution
- Using OpenVINO™ allows developer to maximize HW performance by using common API without having to go to the Metal
- Easy to incorporate deep learning with the Deep Learning Deployment Toolkit
- Trad. And DL are not mutually exclusive

## OpenCL is used for:

- required to run with GPU target (cldnn) using Intel® Processor Graphics
- custom kernels
- other kernels can be used for other non-inference pipeline stages, such as color conversions

Media SDK - API to access intel Quick Sync Video - hw accelerated encoding, decoding and processing

- H.265, H.264, MPEG-2 and more
- Resize, scale, deinterlace, color conversion,, composition, denoise,



sharpen and more

- Outstanding perf., rich API to tune pipeline, support new proc. w/o code change



# 使用模式



# 英特尔® 深度学习部署工具套件






# 步骤 1 — 训练模型

1. 训练后模型将用于模型优化器 (MO)

2. 使用被盗车辆模型训练中的冻结图形 (.pb 文件) 作为输入

3. 模型优化器支持工具将训练后模型转换为冻结图形，以防该任务未完成。





## 步骤 2 — 模型 优化器 (MO)



## 步骤 2 — 模型优化器 (MO)

模型优化器助力提升性能



- 易于使用、基于 Python \* 的工作流程不需要重建框架。
- 从各种框架导入模型 (Caffe \*, TensorFlow \*, MXNet \*, 计划支持更多框架...)。
- 验证了超过 100 个适用于 Caffe\*, MXNet\* 和 TensorFlow\* 的模型。
- 使用标准层或用户提供的自定义层的 模型 IR 文件不需要 Caffe\*。
- 对于不支持的层, 可以回退到原始框架, 但需要原始框架。

The **redesigned Model Optimizer software** is implemented as Python code, replaces the previous solution entirely and offers new features:

- entirely new workflow, at the same time simplified and not requiring User to rebuild Caffe, etc.
- Windows support;
- Caffe is not required to generate IRs for models consisting of Standard Layers, OR when user already provides his custom layers;
- fallback to original framework is possible in case of unsupported layers (then framework is required);
- additional optimizations generalized from existed in the old MO;
- improved usability, stability and diagnostics capabilities; [no analyzer cap]
- total ~110 public models supported for Caffe, MXNet and TensorFlow frameworks – list is available on request;

The Model Optimizer is easier to install, and easier to use for optimizations  
Improved performance and output

深度学习

written in easy Python language, more efficient workflow  
using standard layers, get faster performance without the overhead of frameworks



# 模型优化器助力提升性能（续）

## 模型优化器执行通用优化：

- 节点合并
- 横向融合
- 将 Batch normalization 转换成 scale shift
- 将 scale shift 融合到 convolution
- 终止未用层（dropout）
- FP16/FP32 量化

	FP32	FP16
CPU	是	否
GPU	是	推荐
MYRIAD	否	是
FPGA/DLA	否	是

## 模型优化器可以切断网络的一部分：

- 模型具有无法映射到现有层的预处理/后处理部件。
- 模型有一个在推理过程中不使用的训练部分。
- 模型太复杂，无法一次性转换。

The **redesigned Model Optimizer software** is implemented as Python code, replaces the previous solution entirely and offers new features:

- entirely new workflow, at the same time simplified and not requiring User to rebuild Caffe, etc.
- Windows support;
- Caffe is not required to generate IRs for models consisting of Standard Layers, OR when user already provides his custom layers;
- fallback to original framework is possible in case of unsupported layers (then framework is required);
- additional optimizations generalized from existed in the old MO;
- improved usability, stability and diagnostics capabilities; [no analyzer cap]
- total ~110 public models supported for Caffe, MXNet and TensorFlow frameworks – list is available on request;

The Model Optimizer is easier to install, and easier to use for optimizations  
Improved performance and output

深度学习

written in easy Python language, more efficient workflow  
using standard layers, get faster performance without the overhead of frameworks



# 模型优化器助力提升性能

## 示例

1. 删除批处理规范化阶段。
2. 重新计算权重以“纳入”该操作。
3. 将卷积和 ReLU 合并到一个优化核心。



The **redesigned Model Optimizer software** is implemented as Python code, replaces the previous solution entirely and offers new features:

- entirely new workflow, at the same time simplified and not requiring User to rebuild Caffe, etc.
- Windows support;
- Caffe is not required to generate IRs for models consisting of Standard Layers, OR when user already provides his custom layers;
- fallback to original framework is possible in case of unsupported layers (then framework is required);
- additional optimizations generalized from existed in the old MO;
- improved usability, stability and diagnostics capabilities; [no analyzer cap]
- total ~110 public models supported for Caffe, MXNet and TensorFlow frameworks – list is available on request;

The Model Optimizer is easier to install, and easier to use for optimizations  
Improved performance and output

深度学习

written in easy Python language, more efficient workflow  
using standard layers, get faster performance without the overhead of frameworks



# 处理标准层

- 要生成 IR 文件，MO 必须识别模型中的各层
- 有些层是跨框架和神经网络拓扑的标准层
  - 示例 — 卷积、池化、激活等
- MO 可以轻松为这些层生成 IR
- 使用 MO 的框架特定说明：
  - Caffe:  
[https://docs.openvinotoolkit.org/latest/\\_docs\\_MO\\_DG\\_prepare\\_model\\_convert\\_model\\_Convert\\_Model\\_From\\_Caffe.html](https://docs.openvinotoolkit.org/latest/_docs_MO_DG_prepare_model_convert_model_Convert_Model_From_Caffe.html)
  - Tensorflow :  
[https://docs.openvinotoolkit.org/latest/\\_docs\\_MO\\_DG\\_prepare\\_model\\_convert\\_model\\_Convert\\_Model\\_From\\_TensorFlow.html](https://docs.openvinotoolkit.org/latest/_docs_MO_DG_prepare_model_convert_model_Convert_Model_From_TensorFlow.html)
  - MxNet:  
[https://docs.openvinotoolkit.org/latest/\\_docs\\_MO\\_DG\\_prepare\\_model\\_convert\\_model\\_Convert\\_Model\\_From\\_MxNet.html](https://docs.openvinotoolkit.org/latest/_docs_MO_DG_prepare_model_convert_model_Convert_Model_From_MxNet.html)



# 处理自定义层（可选）

- 自定义层未包含在 MO 已知层列表中。
- 将自定义层注册为模型优化器的扩展
  - 与计算机上的 Caffe\* 可用性无关
- 将自定义层注册为 Custom，并使用系统 Caffe 计算每个 Custom 层的输出形状。
  - 需要系统具备 Caffe Python 接口
  - 要求 CustomLayersMapping.xml 文件明确自定义层
- TensorFlow\* 中的流程也很相似





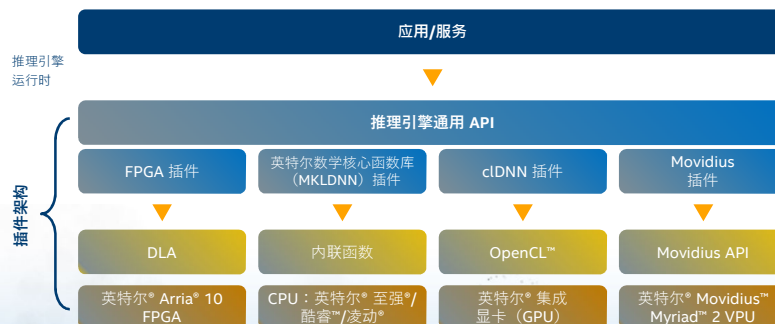
## 步骤3和4—推理引擎和 对CPU/集成显卡/MOVIDIUS/FPGA的支持



# 使用推理引擎实现最佳模型性能

将模型和数据转换为结果和信息

- 简单统一的 API，支持在所有英特尔® 架构 (IA) 上进行推理
- 针对大型人工智能硬件目标的优化推断 (CPU/iGPU/FPGA)
- 异构支持允许跨硬件类型执行层
- 异步执行可提高性能
- 扩展未来英特尔® 处理器的开发，满足未来需求



OpenVX 和 OpenVX 标识是 Khronos Group 的商标。  
OpenCL 和 OpenCL 标识是苹果公司的商标，需获得 Khronos 的许可方能使用

## Heterogeneity – Device affinities.

User can specify (example) "HETERO:FPGA,CPU" to fallback to CPU for layers that FPGA does not support.

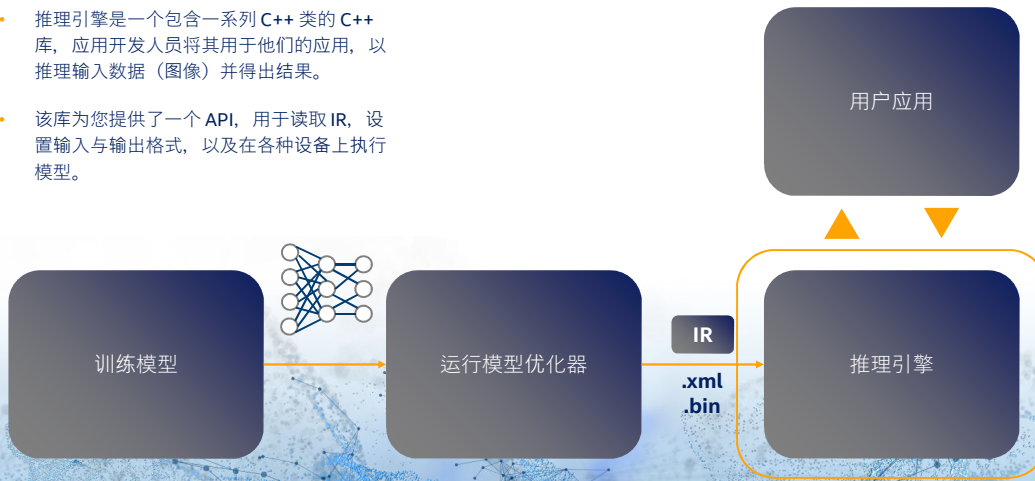
This can also be used for CPU+GPU cases when User has custom layers implemented on CPU only and wants to execute rest of topology on GPU without obligation to rewrite the custom layer for GPU

**Async API** usage improves overall frame-rate of the application, allowing to do other things (like next frame decoding), while accelerator is busy with inference of the current frame.



# 推理引擎

- 推理引擎是一个包含一系列 C++ 类的 C++ 库，应用开发人员将其用于他们的应用，以推理输入数据（图像）并得出结果。
- 该库为您提供了一个 API，用于读取 IR，设置输入与输出格式，以及在各种设备上执行模型。





# 推理引擎插件支持的层

- **CPU — 英特尔® MKL-DNN 插件**
  - 支持 FP32、INT8（已列入计划）
  - 支持英特尔® 至强®/英特尔® 酷睿™/英特尔凌动® 平台 (<https://github.com/01org/mkl-dnn>)
- **GPU — cldnn 插件**
  - 支持 FP32 和 FP16（推荐用于多数拓扑）
  - 支持 Gen9 和上述显卡架构 (<https://github.com/01org/cldnn>)
- **FPGA — DLA 插件**
  - 支持英特尔® Arria® 10
  - FP16 数据类型, FP11 即将推出
- **英特尔® Movidius™ 神经计算棒 — 英特尔® Movidius™ Myriad™ VPU 插件**
  - 英特尔® Movidius™ Myriad™ X 支持一系列层（28 层），不受支持的层必须通过其他推理引擎（IE）插件推理。支持 FP16

Layer Type	CPU	FPGA	GPU	MyriadX
Convolution	Yes	Yes	Yes	Yes
Fully Connected	Yes	Yes	Yes	Yes
Deconvolution	Yes	Yes	Yes	Yes
Pooling	Yes	Yes	Yes	Yes
ROI Pooling	Yes		Yes	
ReLU	Yes	Yes	Yes	Yes
PRReLU	Yes		Yes	Yes
Sigmoid			Yes	Yes
Tanh			Yes	Yes
Clamp	Yes		Yes	
LRN	Yes	Yes	Yes	Yes
Normalize	Yes		Yes	Yes
Mul & Add	Yes		Yes	Yes
Scale & Bias	Yes	Yes	Yes	Yes
Batch Normalization	Yes		Yes	Yes
SoftMax	Yes		Yes	Yes
Split	Yes		Yes	Yes
Concat	Yes	Yes	Yes	Yes
Flatten	Yes		Yes	Yes
Reshape	Yes		Yes	Yes
Crop	Yes		Yes	Yes
Mul	Yes		Yes	Yes
Add	Yes	Yes	Yes	Yes
Permute	Yes		Yes	Yes
PriorBox	Yes		Yes	Yes
SimplerNMS	Yes		Yes	
Detection Output	Yes		Yes	Yes
Memory / Delay Object	Yes			
Tile	Yes			Yes

[https://docs.openvino toolkit.org/latest/\\_docs\\_ie\\_DG\\_supported\\_plugins\\_Supported\\_Devices.html](https://docs.openvino toolkit.org/latest/_docs_ie_DG_supported_plugins_Supported_Devices.html)

Layers in mklDnn and cldnn and extension layers



The image features a blue rectangular background with a subtle, abstract pattern of white and light blue lines and dots, resembling a network or neural structure. Centered on this background is the Intel OpenVINO logo and title in white text.

# 英特尔® OPENVINO™ 工具套件分发版安装



# 安装英特尔® OPENVINO™ 工具套件

- 点击以下链接查看安装说明：<https://software.intel.com/zh-cn/openvino-toolkit/choose-download>
- 遵循 TensorFlow\* 的说明
- 开始前测试一些示例
- 运行推理前，您需要使用模型优化器（MO）将训练获取的冻结图形转换为中间表示







## 使用模型优化器创建 中间表示（IR）文件



# 使用模型优化器生成优化的中间表示（IR）

为 TensorFlow\* 配置模型优化器：

- 使用以下路径的配置 **bash** 脚本（Linux\* 操作系统）或批处理文件（Windows\* 操作系统）为 TensorFlow\* 框架配置模型优化器：

<INSTALL\_DIR>/deployment\_tools/model\_optimizer/install\_prerequisites folder:

install\_prerequisites\_tf.sh

install\_prerequisites\_tf.bat



# 使用模型优化器生成优化的中间表示（IR）

## 转换 TensorFlow\* 模型：

Go to the <INSTALL\_DIR>/deployment\_tools/model\_optimizer directory

- 使用 `mo_tf.py` 脚本轻松将具有该路径的模型转换为具有输出中间表示（即指定 `../models/` 中的 `result.xml` 和 `result.bin`）的输入模型 `.pb` 文件：

```
python mo_tf.py --input_model <TRAIN_DIR>/frozen_inception_v3.pb  
--model_name result \  
--output_dir ../models/
```

- 启动模型优化器支持模型 `.pb` 文件，颠倒 `RGB` 和 `BGR` 之间的通道顺序，并将中间表示的输入和精度平均值指定为 `FP16`：

```
python mo_tf.py --input_model <TRAIN_DIR>/frozen_inception_v3.pb \  
--reverse_input_channels \  
--mean_values [255,255,255] \  
--data_type FP16  
.....
```

**Note:** The Model Optimizer does not revert input channels from RGB to BGR by default, as it did in the 2017 R3 Beta release. Manually specify the command-line parameter to perform this reversion: `--reverse_input_channels`





## 运行时推理



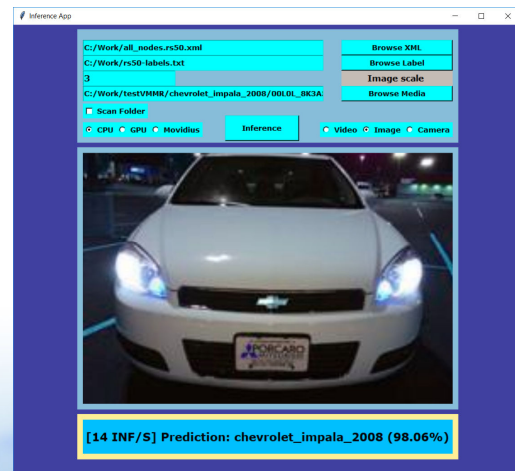
# 边缘推理动手实践 — 教程：

- 简介

- 识别失窃车辆

- 目的

- 本部分将指导用户使用英特尔® 硬件和软件工具开发一种可行解决方案，以创建汽车盗窃分类应用。





# 工作原理

该应用使用了早期练习中的预训练模型。

该模型基于改进的 Inception\_V3 网络，该网络由针对 ImageNet 进行训练的检查点衍生而来，具有 1000 个类别。为实现本练习目的，该模型在最后一层进行了修改，仅能代表最常被盜车辆的 10 个类别。

从 OpenCV 的 VideoCapture 中获取帧后，该应用将使用该模型执行推理。结果显示在带有分类文本和性能数字的帧中。

- 执行推理演示应用请运行：

```
$ python Inference_GUI.py
```



# OPENVINO™ 应用执行流





# 推理步骤

## 1. 加载插件

```
plugin = IEPlugin(device=device_option)
```

## 2. 读取 IR / 加载网络

```
net = IENetwork(model=model_xml,weights=model_bin)
```

## 3. 配置输入和输出

```
input_blob, out_blob = next(iter(net.inputs)),  
next(iter(net.outputs))
```

## 4. 加载模型

```
n, c, h, w = net.inputs[input_blob].shape  
exec_net = plugin.load(network=net)
```

## 5. 准备输入

```
inputs={input_blob: [cv2.resize(frame_, (w, h)).transpose((2,  
0, 1))]}
```

## 6. 推理

```
res = exec_net.infer(inputs=inputs)
```

```
res = res[out_blob]
```

## 7. 处理输出

```
top = res[0].argsort()[::-1]
```

```
pred_label = labels[top[0]]
```



# 在 JUPYTER NOTEBOOK 上运行推理

- 您还可通过 **jupyter notebook** 运行模型优化器以创建 IR 文件（bin/xml），并使用推理引擎进行推理
- 参考第 4 部分 - **OpenVINO\_Video\_Inference.ipynb**
  - 将“arg\_device”参数设置为“CPU”、“GPU”或“MYRIAD”，以在 CPU、集成显卡或英特尔® Movidius™ 神经计算棒上运行







## 在第二代英特尔® 至强® 可扩展 处理器上加速 AI/ML 推理 (INT8)



# 第二代英特尔® 至强® 可扩展处理器



英特尔® 至强® 可扩展处理器上的直接兼容 CPU

## TCO/灵活性

矫健、敏捷地  
踏上人工智能之旅...

- ✓ IMT — 英特尔® 基础设施管理技术
- ✓ ADQ — 英特尔应用设备队列
- ✓ SST — 英特尔® Speed Select Technology

## 性能

借助英特尔® 深度学习加速  
实现内置加速...

提升  
30倍

深度学习吞吐量！<sup>1</sup>  
吞吐量 (图像/秒)

## 安全性

硬件增强的安全性...

- ✓ 英特尔® Security Essentials
- ✓ 英特尔® SecL: 面向数据中心的英特尔® 安全库
- ✓ TDT — 英特尔® 威胁检测技术

<sup>1</sup> 基于英特尔内部测试: 1 倍, 5.7 倍, 14 倍和 30 倍性能提升基于英特尔® 至强® 可扩展处理器上的英特尔® Café ResNet-50 推理吞吐量性能优化。请参阅图表详细描述。3 性能提升基于截至 2017 年 2 月 11 日 (1 倍), 2015 年 11 月 8 日 (5.7 倍), 2019 年 2 月 20 日 (14 倍) 和 2019 年 2 月 26 日 (30 倍) 的测试。可能不会反映所有公开的安全更新。所有产品测试提供绝对的安全性。请参阅英特尔的漏洞披露政策。  
优化声明: 英特尔的编译器针对英特尔处理器的优化程度可能因英特尔处理器型号 (或不) 而不同。这些优化包括 SSE2, SSE3, 和 SSE4.1 指令集以及其他优化。对于在英特尔处理器的微处理器上进行的优化, 英特尔不对性能的可移植性、可用性、性能或性能提供任何保证。英特尔产品中依赖于处理器的优化仅适用于英特尔处理器。英特尔不保证英特尔处理器的优化在所有英特尔处理器上使用。英特尔将提供产品用户指南, 以了解关于性能和优化的特定指令集的其他信息。在性能测试过程中使用的软件及工作负载可能仅针对英特尔处理器的优化进行了性能优化。性能测试 (如 Sysmark R1 MobileMark) 使用特定的测试程序、软件、硬件、固件和/或其他硬件。上述性能提升的奖励和/或可能受到测试程序的变化、测试程序其他版本及性能测试 (包括英特尔产品使用时的运行性能) 以时英特尔产品进行金更评估。英特尔保留更改权利。  
<http://www.intel.com/content/www/ja/jp/benchmarking/intel-product-performance.html>

Let me introduce you to the all new 2<sup>nd</sup> Generation Intel® Xeon® processor scalable family (formerly codenamed Cascade Lake), which is drop-in compatible with the previous Intel® Xeon® Scalable processor platform.

You can use it to:

Achieve the deep learning performance you need thanks to built-in acceleration with Intel DL boost, optimized DL SW frameworks, and the ability to efficiently scale up to hundreds of nodes

Lower TCO/increase utilization by sharing resources between data center and AI workloads, with even more agility thanks to new features like IMT/ADQ/SST

Confidently analyze your sensitive data with hardware-enhanced security including new features like Intel SecL and TDT

And so much more...



# 英特尔® 深度学习加速 (DL BOOST)

采用矢量神经网络指令 (VNNI)



Current AVX-512 instructions to perform INT8 convolutions: vpaddubsw, vpaddwd, vpaddq



Future AVX-512 (VNNI) instruction to accelerate INT8 convolutions: vpdpbusd\*\*

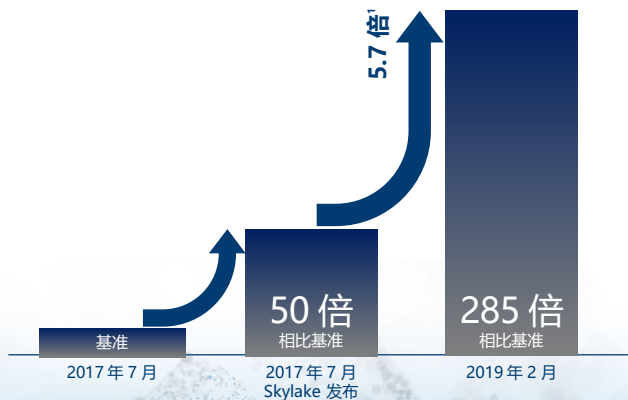


Intel® Deep Learning Boost (VNNI) on the 2<sup>nd</sup> Generation Intel® Xeon® Scalable processor is designed to deliver significant, more efficient Deep Learning (Inference) acceleration.

- Intel® DL Boost (VNNI) is a new Intel® Advanced Vector Extension (Intel® AVX-512) instruction
- It is a fused multiply-add instruction, which is often used in matrix manipulations as part of deep learning inference
- The new VNNI instruction combines what were three separate instructions into a single processor instruction, saving clock cycles on the processor.
- VNNI can help to speed up image classification, speech recognition, language translation, object detection and more



## 英特尔®至强®处理器的硬件 + 软件改进




Framework	Model	Speedup	GPU
Caffe	Resnet-50	1.9x <sup>2</sup>	V
	Inception v3	1.8x <sup>2</sup>	
	SSD-VGG16	2.0x <sup>2</sup>	
TensorFlow	Resnet-50	1.9x <sup>2</sup>	N
	Inception v3	1.8x <sup>2</sup>	I

[illegible]

129

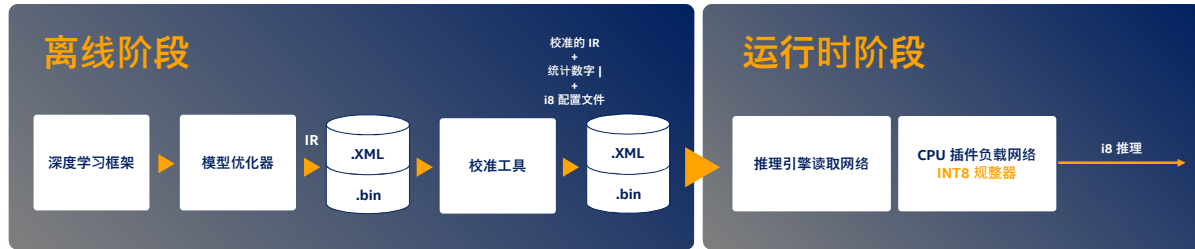




# 使用英特尔® OPENVINO™ 工具套件分发版实施 INT8 推理



# workflow



· 工作流程类似于 Fp32，将“校准工具”用于 INT8 除外。



# 转换训练后模型和推理的步骤

OpenVINO 工具套件支持在英特尔处理器上实施 int8 模型推理：

- 使用**模型优化器工具**(<https://software.intel.com/en-us/articles/OpenVINO-ModelOptimizer>)  
对模型的原始框架格式进行转换，以中间表示（IR）格式输出模型。
- 使用英特尔 OpenVINO 工具套件分发版中的**校准工具**  
([https://docs.openvino toolkit.org/R5/\\_samples\\_calibration\\_tool\\_README.html](https://docs.openvino toolkit.org/R5/_samples_calibration_tool_README.html))  
实施模型校准。它接受模型的 IR 格式，且不受框架影响。
- 使用 IR 格式的更新模型实施推理。





# 课程结业证书



# 课程结业证书

- 您可以选择在完成课程测验时获得英特尔® 人工智能课程结业证书。
- 开始测验前，您可能需要禁用广告拦截工具。（Ghostery、uBlock、AdGuard 等）
- **参加测验:** [https://intel.az1.qualtrics.com/jfe/form/SV\\_9EIVi2JXNF1ViiV](https://intel.az1.qualtrics.com/jfe/form/SV_9EIVi2JXNF1ViiV)







# 详细了解 英特尔的 AI 产品



# 资源

- 英特尔® OpenVINO™ 工具套件分发版
- <https://software.intel.com/en-us/openvino-toolkit>
- 强化学习 Coach
- <https://github.com/NervanaSystems/coach>
- NLP 架构师
- [http://nlp\\_architect.nervanasys.com/](http://nlp_architect.nervanasys.com/)
- Nauta
- <https://www.intel.ai/introducing-nauta/#gs.8hTP6kBc>
- BigDL
- <https://software.intel.com/en-us/ai/frameworks/bigdl>
- 面向 Caffe\* 的英特尔优化
- <https://software.intel.com/en-us/ai/frameworks/caffe>
- 面向 TensorFlow\* 的英特尔® 优化
- <https://software.intel.com/en-us/ai/frameworks/tensorflow>

更多信息请关注人工智能网络研讨会系列:

<https://software.seek.intel.com/AIWebinarSeries>

## 人工智能课程

- 人工智能简介
  - <https://software.intel.com/en-us/ai/courses/artificial-intelligence>
- 机器学习
  - <https://software.intel.com/en-us/ai/courses/machine-learning>
- 深度学习
  - <https://software.intel.com/en-us/ai/courses/deep-learning>
- Tensorflow\* 的应用深度学习
  - <https://software.intel.com/en-us/ai/courses/tensorflow>



