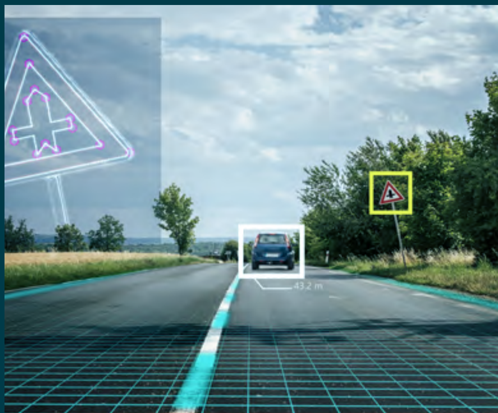






Data-Centric Computer Vision

Approaches, Applications, and Best Practices

An Appen eBook

appen

Table of Contents
Data-Centric Computer Vision

Introduction

The world of computer vision is full of inputs that come in all forms of text, images, and video. Just like machine learning (ML) algorithms have enabled us to extract information from text, recent developments in the field of Computer Vision (CV) have made it possible for algorithms to glean insight from image and video as well. **CV, a subdomain of artificial intelligence (AI), seeks to teach computers to "see" in the way humans do in order to understand the content of digital media.**

Unlike an ML algorithm, our human vision systems have the advantage of learning from a lifetime of experiences how to contextualize the things we see. To equip computers with the ability to contextualize, we must provide them an enormous amount of real-world scenarios to learn from. These examples may come in many forms: 2D images and video (taken from an SLR or infrared camera), 3-D images and video (taken from a camera or scanner), and sensor data (taken from RADAR or LiDAR technology). In any case, high-quality data is foundational to effective CV systems.

With the introduction of deep learning and neural network techniques, the field of CV is experiencing unprecedented growth. Today, a Computer Vision system can recognize objects in images and video, including their shapes, textures, colors, sizes, locations, movements, and other relevant characteristics. With these abilities, CV unlocks tremendous opportunities for speed, efficiency, and growth in business operations around the globe.

This eBook is focused on a data-centric approach to AI development, which consists of systematically changing and enhancing datasets to improve the accuracy of AI system, as opposed to adjusting the models.

In this guide, we'll walk through CV approaches and applications, sharing techniques and best practices for launching successful CV-based models.

Factors Driving Growth In Computer Vision

Surprisingly, Computer Vision has been around in some form since the 1960s. Back then, it was thought that simply attaching a camera to a computer would enable it to “see” in the way humans do (spoiler: that didn’t work). CV is much more complicated now due to advances in machine learning, but also exceedingly more useful thanks to a variety of factors that have helped unlock untapped potential in the field:

Data

Using cameras, sensors, scanners, and innovative new imaging technology, there are plenty of opportunities to create more data for CV projects. The expansion of data storage options, particularly with regards to the cloud, has likewise increased the availability of data. In 2011, the average company spent $6.5k for the entire year on Cloud data storage, compared to 2020, when that number increased to $10k per month. By 2023, researchers predict that the majority of enterprises will store their data on the Cloud. If we look at the sheer amount of data being produced, an impressive 44 zettabytes (that’s 21 zeroes) was available at the start of 2020, with 463 exabytes (that’s 18 zeroes) expected to be generated each day by 2025.

Imaging Devices

Technological advancements in image-capturing hardware and software provide greater accuracy for image analysis. For example, smartphones, scanners for capturing LiDAR data, alternate reality devices, and drones can all provide increasingly high-quality images. Image devices like these are more affordable and available than ever, lending to the capture of more data for CV projects.

Processing Power

There has been significant improvement in the last few years in the ability to process large volumes of data quickly and effectively.

Annotation Tools

Organizations have at their disposal a growing selection of annotation tools available for data labeling purposes, improving process speeds and driving scalability. There are several open-source annotation tools available, as well as third-party vendors that provide access to their own annotation platforms. In any case, there are numerous options for annotating images and video, and selecting one over another would depend on the use case and annotation techniques required for the project.

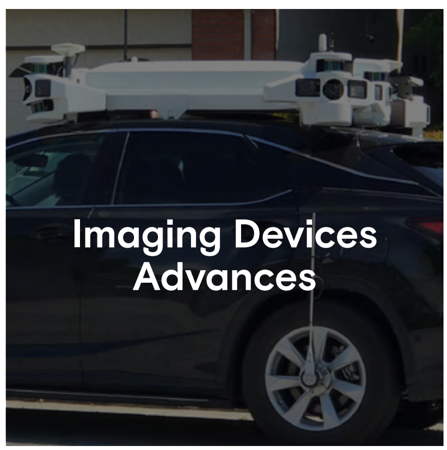
Cost

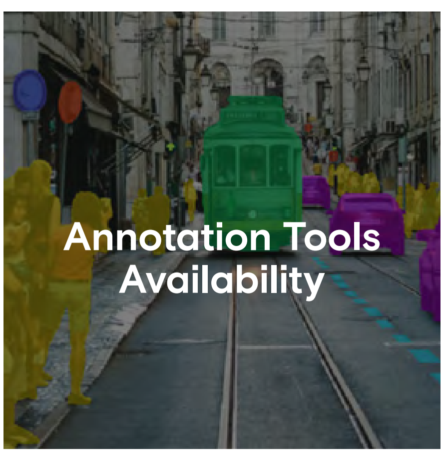
The cost of computing continues to drop exponentially. In some estimates, computer power available per dollar has increased by tenfold every four years in the last 25 years.

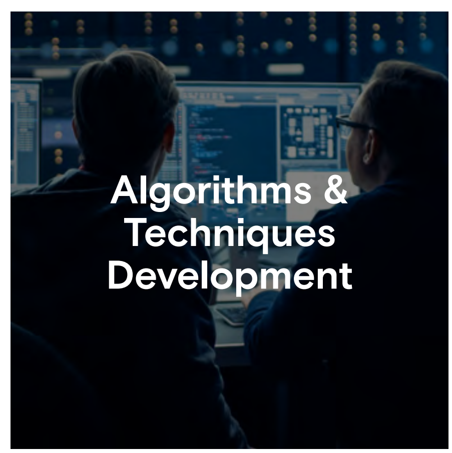
Algorithms and Techniques

CV has many subareas such as object detection, segmentation, and object tracking. The recent introduction of deep neural networks has accelerated the development pace of many if not all CV areas to solve real-world problems. In addition, open source machine learning models encourage more rapid experimentation across multiple industries.

With the developments outlined above, Computer Vision is ripe for investment. Indeed, many industries have already hopped on the CV train to tap into its potential for creating cost-saving efficiencies and new sources of revenue.

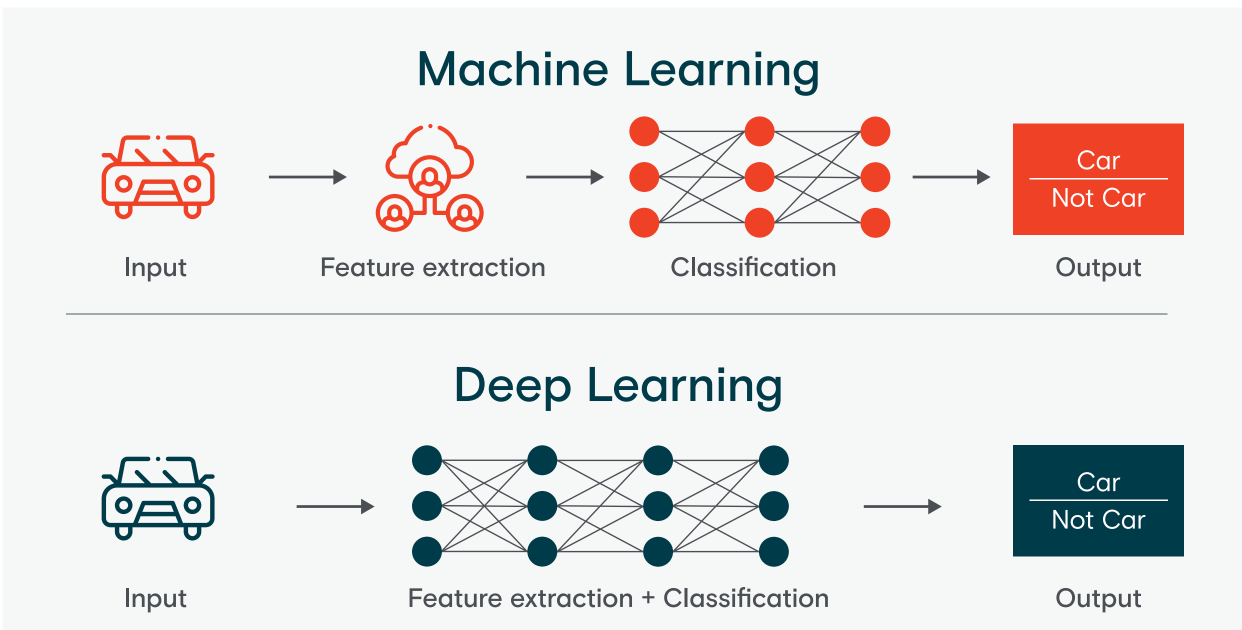






Approaches to Computer Vision

Computer Vision may have been around for decades, but applying machine learning to the discipline is what launched it to the forefront of innovation. While deep learning (a subdomain of ML) is considered the cutting edge technique in CV, there are other approaches that can work, depending on your use case. Comparing two of the most popular approaches demonstrates the efficiencies offered by deep learning and why this approach has taken center stage.



The Standard Machine Learning Approach

A non-deep learning approach typically requires engineers to handcraft visual features to be used in their CV tasks. In this approach, humans explicitly tell the algorithm what to look for.

The Deep Learning Approach

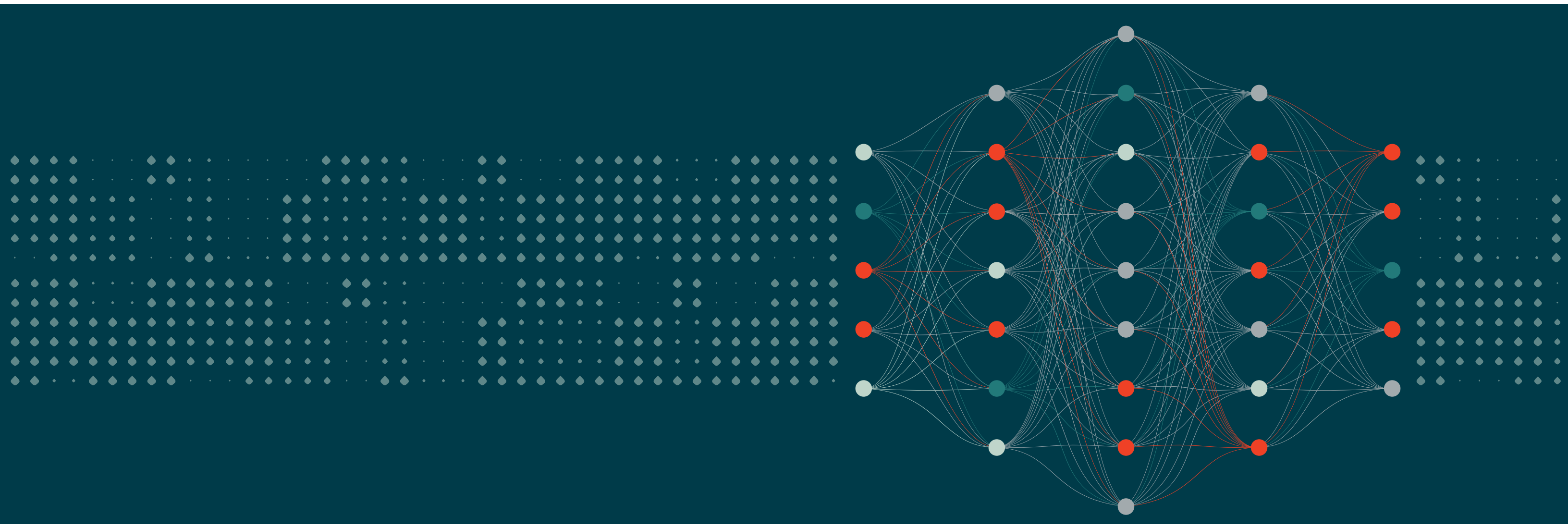
Deep learning algorithms rely on artificial neurons in a network such as Convolutional Neural Networks (CNNs) to learn their connections to other neurons. In this approach, visual features are automatically learned from input data instead of hard-coded by engineers. It has been shown in many CV areas that this approach can outperform traditional ML algorithms and even achieve super human accuracy.

How Neural Networks Work

Modern artificial neural networks require enormous amounts of high-quality data and repeated iterations of training for e.g., turning the connection weight between two neurons. They're meant to mirror the human learning process: review data, identify patterns or relevant features, and use those patterns to create rules for processing new data. Let's look at a simple example. Assume we're teaching a deep learning algorithm to recognize whether cars are present in a given image.

1. **Provide the algorithm with two datasets**: one with only images of cars (and labeled as such), and one with images that contain no cars.
2. **The CNN compares the datasets,** extracting features and patterns to create rules. These rules help the network understand what each image set has in common and how they differ.
3. **Using these rules**, the CNN produces a deep learning model that can isolate car features. In other words, the model can identify if an image contains a car based on whether the features in the image match the features in the algorithm's definition of a car.
4. **The model is then evaluated for accuracy** using a set of images that has not been part of the initial training set.

In practice, the process is much more complicated, but this provides a useful illustration of what's happening at the base level. Due to no feature engineering required and being easier to transfer one solution to other projects without losing quality, data scientists increasingly rely on deep learning techniques for CV projects over the standard machine learning approach. Despite less human supervision, deep learning algorithms have proven to produce high-confidence, accurate results compatible with the requirements for successful CV projects.



Annotating a Computer Vision Project

If there's one thing that all Computer Vision projects have in common, it's that they require huge amounts of high-quality data. How much data you'll need, exactly, will be dependent on the complexity of the problem you're trying to solve. You may collect data from internal sources, open-source datasets, or other third parties; regardless of where you source your data, you need to be sure of three things:

Your data is

Protected and secure - especially if it contains personal information.

Complete - it covers all of your use cases and edge cases.

Clean - if not, you'll need a process in place to remove erroneous, low-quality data.

Assuming that the above conditions are met, you're likely ready to think about annotation. In this stage, ask yourself the following questions:

Who will annotate my data? Given the enormous number of images or video files you'll need to annotate, you can imagine that you'll need a lot of annotators. You ideally don't want your data scientists to spend their time annotating when they could be working on more strategic endeavors and you may not have enough other individuals in your organization who have the time required. Many organizations turn to outside help to handle these challenges, but it's important to choose the right partner wisely, since some tasks are more complex than others.

You can hire temporary contractors to manage data annotation, as this is less expensive than hiring full-time employees. You may also turn to a data provider who can give you access to a crowd of annotators, usually sourced from different parts of the globe. The benefit to you is instant access to a large number of annotators, a diversity of perspectives, as well as expertise. LiDAR annotations, for example, are very complex, and the data annotators' expertise can save time and cycles.

Will I include any automation in my annotation process? Traditionally, humans have manually labeled all of the data in a machine learning project. Now, we have tools that help automate at least a portion of this process. For example, you can:

- Use a pre-trained model to make an initial hypothesis of your data labels. Human annotators will then check that hypothesis for accuracy.
- Use a model to assist with labeling during the job; for instance, by providing object detection across frames in video annotation, or a magic wand tool to help with pixel-level selection in images.
- Use a model to validate a human annotator's label prior to submitting the job, as a form of quality assurance.

These examples show the various touch points where you can employ machine learning tools for greater speed, efficiency and quality compared to fully manual processes.

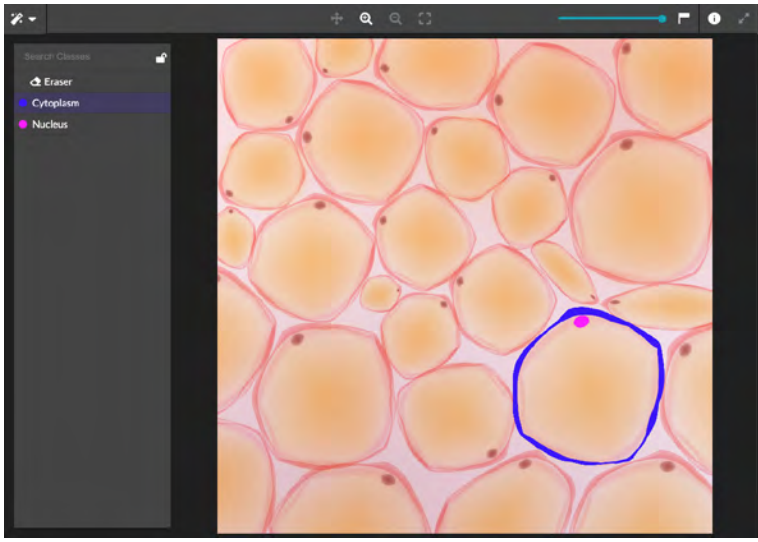
If you choose a third-party data tool for your annotation needs, check that its platform includes the following features: dataset management, annotation methods, data quality control, workforce management, security, integrated labeling capabilities - and the ability to annotate large volumes of images at scale, efficiently.

How will I know if my data is annotated accurately? It's important to have metrics to measure the accuracy of your labels and processes to validate them. One way to do this is via the methods outlined in the previous question, where an ML model verifies human judgments. Another way is to ensure that one or more people review each label for accuracy. You can even use a combination of methods.

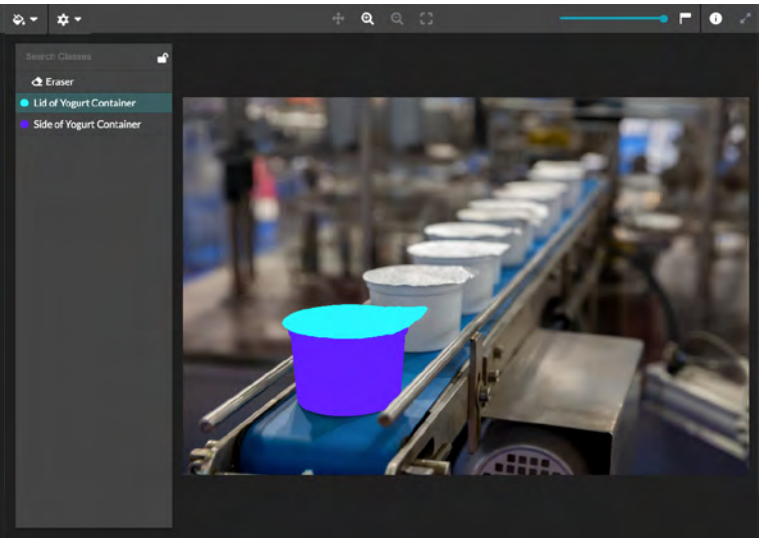
No matter the process you select, be sure you have a dashboard that tracks the accuracy levels. Many third party data providers have key performance indicator dashboards built into their platforms.

How will I know if there's bias in my data annotations? First, ensure that you have a diversity of people annotating your data. These people should represent the various backgrounds and experiences that your end users will potentially have. Second, include in your KPIs metrics that measure bias specifically. A human-in-the-loop can randomly sample labels to check them for bias as well.

What annotation technique(s) will I use? Depending on whether you're doing an image or video annotation project, you have several annotation techniques available to you. For image annotation, you can use classification, different types of object detection, and/or forms of segmentation. With video, the single frame method and the much more efficient continuous frame method are available options. You may want to use a combination of these techniques, or apply different phases throughout your project. In this section, we'll review the key annotation techniques and their use cases.



Example of a Pixel Level Semantic Segmentation job annotating a cell.



Example of a Pixel Level Semantic Segmentation job annotating a yogurt cup.

Example of output from a
Pixel Level Semantic Segmentation job:

```
[
  {
    “description”: “Cytoplasm is depicted as pink
    shaded wall around the cell.”,
    “class_name”: “Cytoplasm”,
    “display_color”: “#3b18f7”
  },
  {
    “description”: “Nucleus is depicted as small
    round matter within the cell.”,
    “class_name”: “Nucleus”,
    “display_color”: “#f71fff”
  }
]
```

Example of output from a
Pixel Level Semantic Segmentation job:

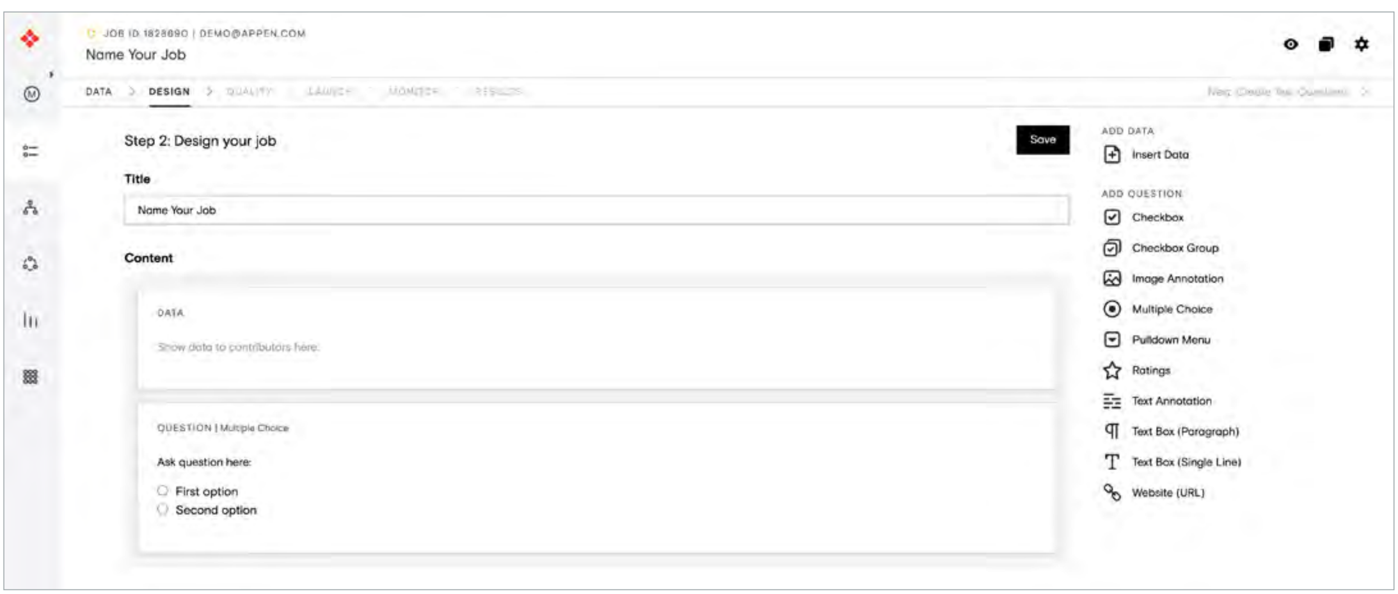
```
[
  {
    “description”: “The foil top of the yogurt
    container is the lid.”,
    “class_name”: “Lid of Yogurt Container”,
    “display_color”: “#17f7ff”
  },
  {
    “description”: “The remaining plastic
    permieter is the side of the yogurt
    container.”,
    “class_name”: “Side of Yogurt Container”,
    “display_color”: “#621fff”
  }
]
```

Image Annotation

In image annotation, annotators add metadata or tags to image data to identify the features the model needs to learn to recognize. The algorithm processes these labeled images, learning from these examples how to identify the features in fresh, unlabeled data. As technology advances, annotators have been able to rely more on Machine Learning assisted features often built into data annotation tools which help automate parts of the annotation processes, increasing efficiency and accuracy. These Machine Learning assisted features also reduce cognitive load on the annotator and improve their annotating experience.

Some of the Machine Learning assistance we have in the Appen Data Annotation Platform include:

- **Pre-Labeling for Semantic Segmentation of Street Scenes** - this increases contributor productivity by up to **91.5%** and annotation quality by up to **10%**.
- **Optical Character Recognition (OCR)** transcription which can be **5x faster** than transcribing images by hand.
- **Speed Labeling for our Video Object Tracking** which increases annotation time by up to **100x**.
- **Speed Labeling for LiDAR** annotations which increases annotation time by up to **4x**.
- **Machine Learning Assisted Lane Line Segmentation** which can be **6x faster** when compared to the manual alternative.



A view of the Job Design section in the Appen Data Annotation Platform

Different types of image annotation exist for different purposes, as highlighted below:

Classification

Image classification asks the question, what kind of object is in this photograph? It's the easiest, most basic method of image annotation because it only applies one tag to an image. It doesn't attempt to localize the object within the image, only to ascertain whether the object is there.

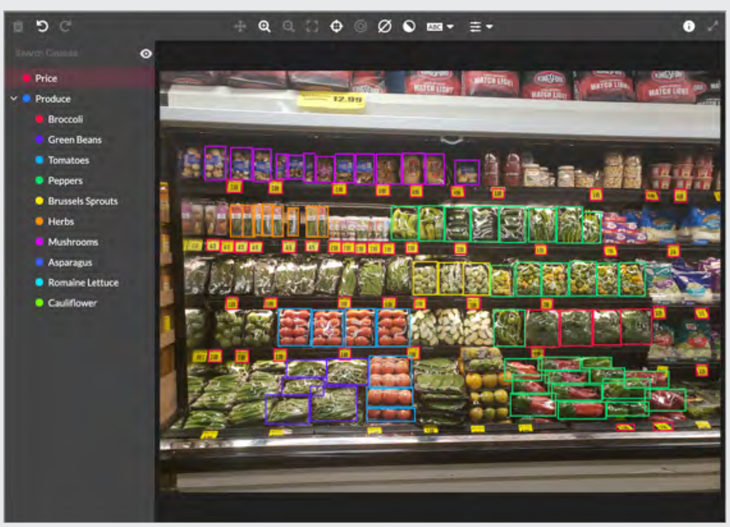
As simple as it sounds, this can be a helpful first step for filtering before launching into more intensive annotation work. If you're training your algorithm to identify crops in images of farmland, you'll want a wide variety of images that contain crops, as well as many that don't (if you have too many of one type, your algorithm will become biased toward the more represented data). Classification helps you understand how much data you have of each type and filter out any unnecessary units of data. The limitation of classification is that it's vague and only useful for simple tasks.

Object Detection

Object detection asks, where are the objects in the image? It provides the target object's general location using a number of annotation techniques:

2D Bounding Boxes

A bounding box is a rectangle or square that serves as a point of reference for object detection. Annotators apply bounding boxes to define the x and y coordinates of the target objects in an image.



An example of an object detection job showing different vegetables being detected.

Here is an abridged sample of the bounding box output data, showing a few of the different classes annotated:


```
[
  {
    "average_trust": 1,
    "class": { "Price": 1 },
    "coordinates": { "h": 85, "w": 92, "x": 1555, "y": 8 },
    "type": "box"
  },
  {
    "average_trust": 1,
    "class": { "Broccoli": 1 },
    "coordinates": { "h": 264, "w": 243, "x": 2862, "y": 1716 },
    "type": "box"
  },
  {
    "average_trust": 1,
    "class": { "Green Beans": 1 },
    "coordinates": { "h": 224, "w": 338, "x": 1083, "y": 2266 },
    "type": "box"
  }
]
```

3D Point Cloud Annotation

If you're working with RADAR or LiDAR data, you'll have a 3D model of the image. A 2D bounding box wouldn't work accurately in this case, so annotators must use 3D bounding boxes, or cuboids. These cuboids, when applied, localize the target object and indicate its depth. This is also known as point cloud annotation.

Handling Multi-Modal Data - Autonomous Vehicle Use Cases

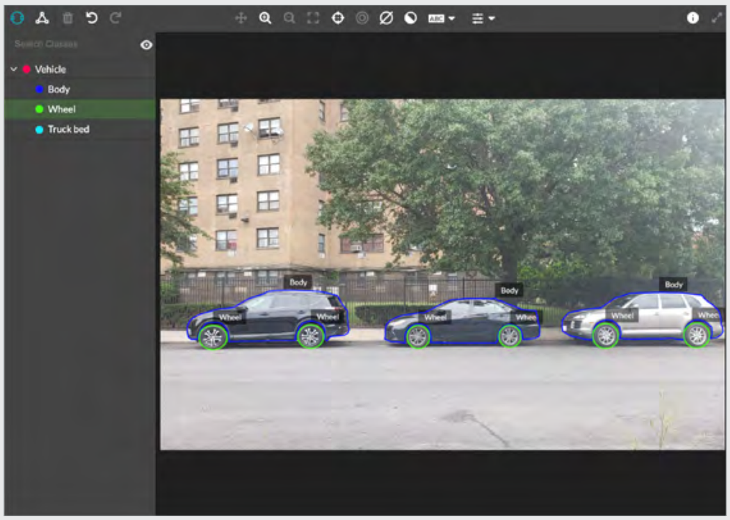
High-quality training data is essential to ensuring autonomous vehicles operate safely, but combining multiple datasets together from several sensors can be challenging without the right tools. At Appen, we work with 7 of the 10 largest global automotive companies and tier 1 suppliers, and can deliver 99+% accuracy for highly complex multi-modal AI projects with the most advanced machine learning-powered data annotation platform.

> "Our project is still in the pilot phase, and we needed to speed up the cycle to reach production, which requires training data that rapidly meets our algorithm requirements. The annotation tool, including 3D LiDAR, high-quality control features, and workflows, is already built into the Appen platform. This is helping us ensure the process is optimized based on our project requirements, enabling a smooth collaboration between our team and the Appen team. We are looking forward to moving this internal pilot into production."

Senior Project Leader at Ecarx,
an automotive technology company building an intelligent, connected platform for multiple vehicle models.

Polygonal Segmentation

In many cases, target objects are asymmetrical and don't fit easily into a box. In this case, or in the case where teams want more precision, annotators use polygonal shapes to define target object locations. This is known as polygonal segmentation.



In this picture, polygons and ellipses are used to annotated the body and wheels of cars for more detailed annotations compared to using bounding boxes.

Here is an abridged sample of polygon and ellipse annotation data. This polygon data shows a subset of the full set of coordinates, while the ellipse data here is one complete ellipse.


```
[
  {
    "average_trust": 1,
    "class": { "Body": 1 },
    "coordinates": [ { "x": 256, "y": 1691 },{ "x": 214, "y":
1691 },{ "x": 193, "y": 1676 },
  ],
    "type": "polygon"
  },

  {
    "average_trust": 1,
    "class": { "Wheel": 1 },
    "coordinates": { "rx": 99, "ry": 86, "x": 367, "y": 1678 },
    "type": "ellipse"
  }
]
```


Landmarks

Mainly used for facial recognition, landmark annotation can identify human body parts and estimate their positions by placing points, or dots, on an object. Other use cases also come up in the fitness and personal training space, as well as gaming and AR/VR applications where body tracking is needed.

Lines and Splines

For some images, the target object may be a line or curve (especially the case in labeling roadways for autonomous vehicle use). Annotators can label lines and splines (curves) to separate key boundaries and regions.

Overall, object detection still isn't the most precise way of localizing objects in images. While it can provide a general location, the shapes (especially bounding boxes) will never be exact and can overlap. Still, it can be an efficient annotation method for projects that don't require high levels of precision.

Ontology Attributes for Image Annotation

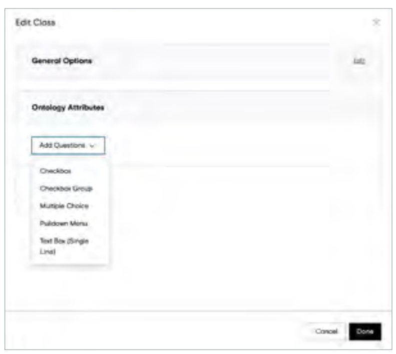
Leveraging Ontology Attributes allows you to collect richer data in with the right image annotation tools (e.g. whether a vehicle is occluded or truncated). An advanced tool will allow you to create ontology attributes specific to your use case.

By categorizing images and assigning attributes in a single tool, labeling is more streamlined and efficient.

1

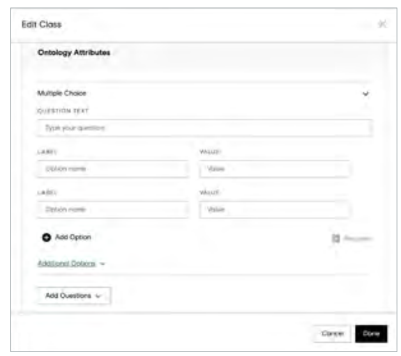
Selecting a class to edit via the Ontology Manager

2

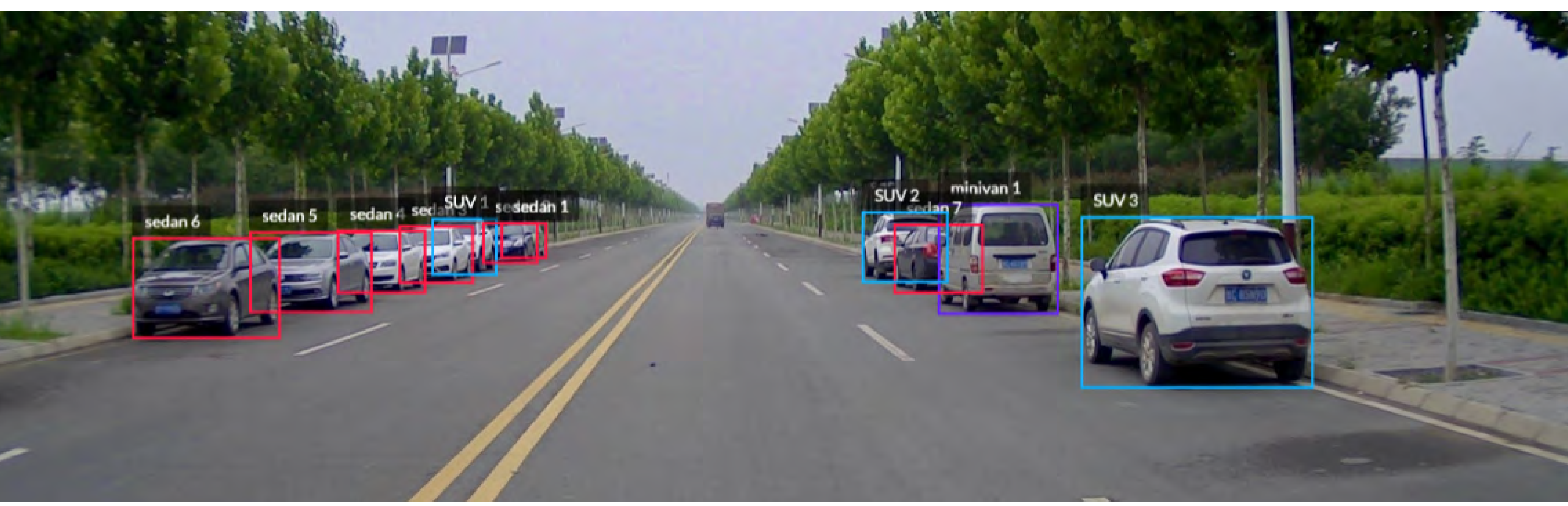


Ontology Attributes section and question type options

3



Example question set-up with Ontology Attributes



ADAP Box Annotation with Ontology Attributes

Ontology Attributes Output

Within the `metadata` object per shape, there are `shapeAnswers` and `shapeQuestions` arrays to represent the attributes labeled for an object and information on the ontology available to the contributor.

Example shape annotation output from an ontology attributes job:


```
{
  "annotation": [
    {
      "id": "25fb3389-4c23-420d-a989-bdfae8a46a5e",
      "class": "sedan",
      "number": 1,
      "type": "box",
      "coordinates": {
        "x": 620,
        "y": 520,
        "w": 53,
        "h": 41
      },
      "metadata": {
        "shapeAnswers": [
          {
            "type": "Checkbox Group",
            "customUserId": "sedan_occlusion",
            "name": "is_the_sedan_occluded",
            "answer": {
              "values": [
                "yes"
              ],
              "customUserId": [
                "sedan_occlusion_yes"
              ]
            }
          }
        ],
        "shapeQuestions": [
          {
            "id": "e817e005-adeb-44f7-8255-ad2d393cd7e4",
            "customUserId": "sedan_occlusion",
            "type": "Checkbox Group",
            "data": {
              "questionText": "Is the sedan occluded?",
              "questionChoices": [
                {
                  "id": "cd3405e7-5123-4053-8a33-
                        e0edda416272",
                  "customUserId": "sedan_occlusion_yes",
                  "label": "Yes",
                  "value": "yes"
                },
                {
                  "id": "16096354-b0dc-444c-9f56-
                        cecc9c204867",
                  "customUserId": "sedan_occlusion_no",
                  "label": "No",
                  "value": "no"
                }
              ],
              "isRequired": true,
              "tipsHints": "Occlusion means the
                            visibility of the car
                            is obstructed.",
              "resultsHeader": "is_the_sedan_occluded"
            }
          }
        ]
      }
    }
```

Instance vs Semantic Segmentation

Semantic segmentation treats multiple objects within a single category as one entity – objects shown in an image are grouped based on defined categories, like in a street scene, where it would be segmented by “pedestrians,” “bikes,” “vehicles,” “sidewalks,” etc.

Instance segmentation, on the other hand, identifies individual objects (or ‘instances’) within these categories. For example, a category like ‘sedans’ will be split into the individual objects, sedan 1, sedan 2, sedan 3 etc to respect the different instance identities.

Instance Segmentation

Semantic Segmentation

Instance segmentation output example for annotated cars:

```
[
  {
    “description”: “Anything one might buy to drive
as a personal vehicle (other than ego car). Car, jeep,
SUV, van with continuous body shape, caravan, no other
trailers.”,
    “class_name”: “car”,
    “display_color”: “#00008e”,
    “children”: [
      {
        “description”: “”,
        “class_name”: “car 1”,
        “display_color”: “#FF1744”
      },
      {
        “description”: “”,
        “class_name”: “car 2”,
        “display_color”: “#651FFF”
      },
      {
        “description”: “”,
        “class_name”: “car 3”,
        “display_color”: “#00B0FF”
      }
    ]
  }
]
```

Semantic segmentation output example for annotated cars:

```
[
  {
    “description”: “Anything one might buy to drive
as a personal vehicle (other than ego car). Car, jeep,
SUV, van with continuous body shape, caravan, no other
trailers.”,

    “class_name”: “car”,
    “display_color”: #00008e”
  }
]
```

Pre-Labeling for Semantic Segmentation

Machine Learning provides an initial ‘best guess’ hypothesis before contributors start the task. With human contributors reviewing pre-processed annotations instead of starting a judgment from scratch, the time needed to annotate data drastically reduces.

Unlabeled picture of street with cars, trees, etc.

PLSS pre-labeled with model predictions showing different classes of cars, trees, road etc.

This is an excerpt of the output data from this pre-labeled PLSS job.

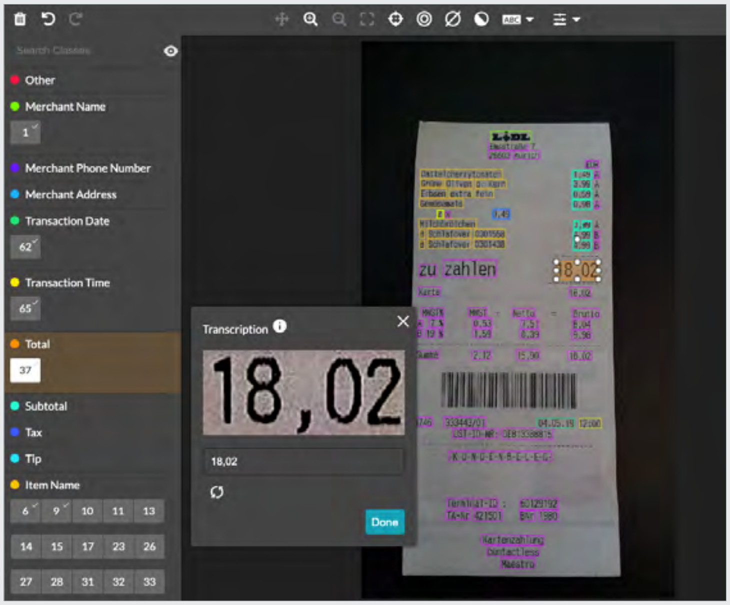

```
[
  {
    “description”: “Anything one might buy to drive as a personal vehicle (other than ego car). Car, jeep, SUV, van
with continuous body shape, caravan, no other trailers.”,
    “class_name”: “car”,
    “display_color”: “#00008e”
  }, ...
  {
    “description”: “Part of ground on which cars usually drive, i.e. all lanes, all directions, all streets.
Including the markings on the road. Areas only delimited by markings from the main road (no texture change) are
also road, e.g. bicycle lanes, roundabout lanes, or parking spaces. This label does not include curbs.”,
    “class_name”: “road”,
    “display_color”: “#804080”
  },
  {
    “description”: “Part of ground designated for pedestrians or cyclists. Delimited from the road by some obstacle,
e.g. curbs or poles (might be small), not only by markings. Often elevated compared to the road. Often located
at the sides of a road. This label includes a possibly delimiting curb, traffic islands (the walkable part), or
pedestrian zones (where usually cars are not allowed to drive during day-time).”,
    “class_name”: “sidewalk”,
    “display_color”: “#f423e8”
  }
]
```

Image Transcription

Sometimes images contain text that needs to be captured and then transcribed. While annotators can do this manually through the use of bounding boxes and text input fields, automated tools are much faster.

Optical Character Recognition (OCR)

OCR is an automated form of transcription, wherein an algorithm can convert images of typed, handwritten or printed text into machine-encoded text. This method is more efficient and thanks to recent machine learning advances, more accurate than the manual alternative.



This image shows a screenshot of a Machine Learning Assisted Image Transcription job.

This is an excerpt of the output data from the image transcription job shown on the left.


```
[
  {
    {"id":"27c39b4b-f489-409a-808d-533d7e4e3588",
    "class":"Total",
    "metadata:[
    {
        "label":"Transcription",
        "modelType":"ocr",
        "modelDataType":"block",
        "inputType":"digit",
        "annotatedBy":"machine",
        "text":"18,02",
        "confidence":0.96
    }
    ],
    "instance":37, "type":"box", "angle":0,
    "coordinates" : {" x":396, "y":449, "w":88, "h":37}
  }
]
```

Video Annotation

Video annotation assists your models with seeing and interpreting movement. To gain the benefits of video annotation over image annotation (namely, it's cost- and time-savings), you'll need to incorporate some level of automation. Many third parties offer video annotation automation tools that address specific use cases.

Single Image

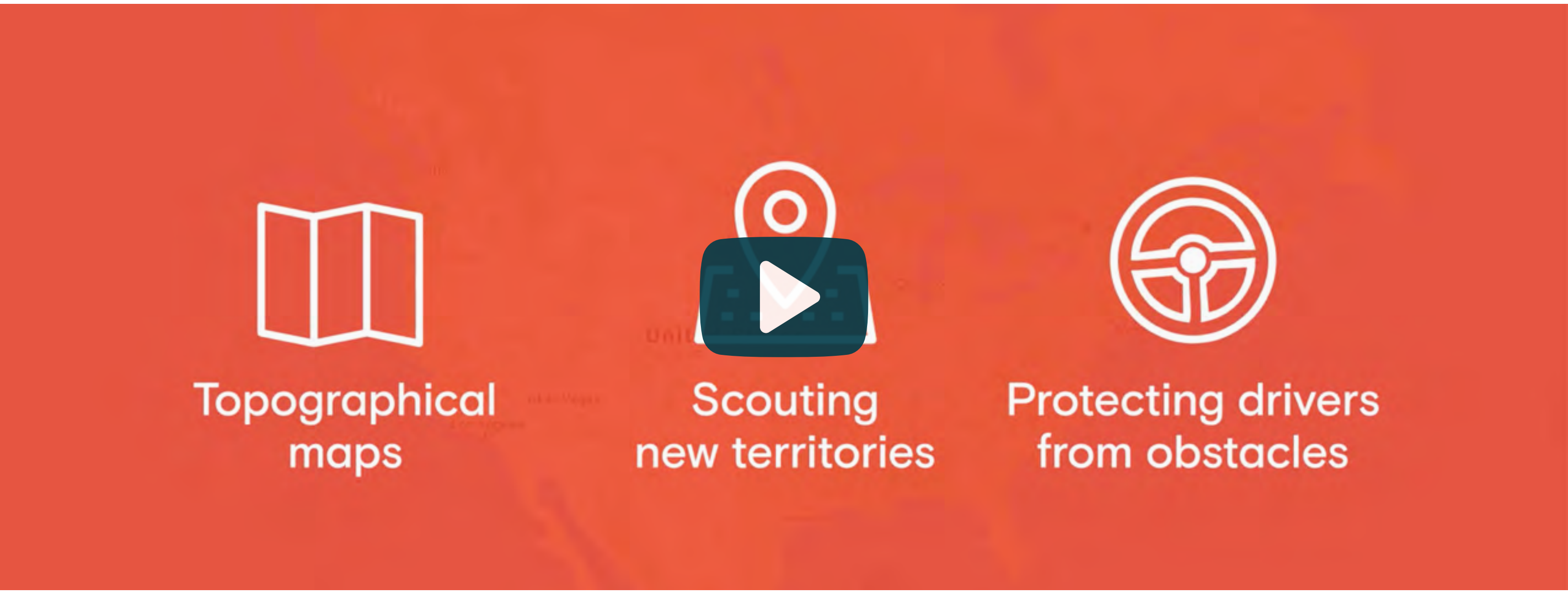
To illustrate the power of automation, the single image method is what happens when video annotation must be completed manually. In this method, teams extract all frames from a video and then annotate them as images, using the image annotation techniques described previously. A video at a standard speed of 30fps could include nearly 2,000 frames per minute, adding up to huge amounts of annotation time.

This method is costly and error-prone, as labelers could misidentify the same object from one frame to the next.

Continuous Frame

Fortunately, the continuous frame method uses automation to provide all of the benefits that video annotation offers. In this method, automation tools perform object tracking to automatically label objects and their locations in subsequent frames. The human labeler need only label the object once, the first time it appears, and the computer can do the rest of the work, with the labeler adjusting annotations if needed.

Automation provides greater consistency, reducing opportunity for labeling errors.



Video Transcription

In video transcription, annotators or machine learning tools can transcribe audio or annotate motion. They can timestamp specific areas and tag events as well to enhance transcription accuracy.

Should I use images or videos for my Computer Vision project?

In part, this depends on the nature of the problem you're trying to solve; in some cases, using images versus video may be a requirement you can't avoid. If you're able to compare the two, though, video annotation generally offers more benefits:

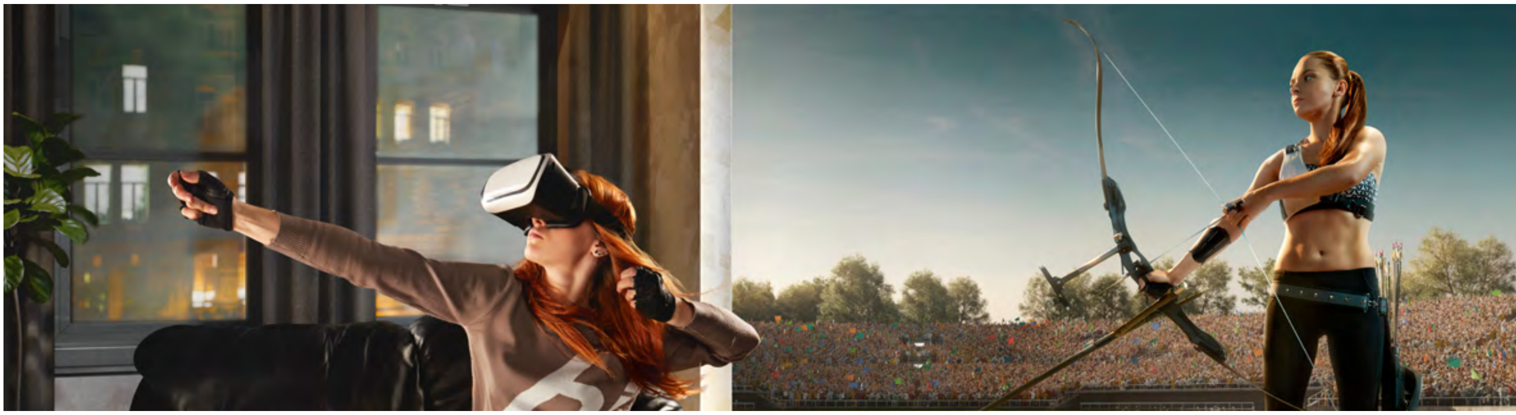
- **Partial automation makes it a faster process**
- **Continuity between frames reduces chance for error**
- **More information per unit of data: video provides movement and audio**

Real-life Applications

Almost every time you order a package, enter a store, or watch a movie with CGI, you're interacting with Computer Vision. CV solutions are more ubiquitous in our daily lives than we probably realize; that's because every major industry has incorporated CV into their business operations. Here are a few highlighted use cases to illustrate the power of Computer Vision and how its annotation techniques work in practice:

Augmented Reality / Virtual Reality (AR/VR)

AI models can offer an easier and more scalable way to provide the inputs required to build seamless and impressive AR/VR experiences. The most common way is by recognizing data inputs and triggering an effect within the AR/VR scene. For example, using Image and Scene Labeling to classify an image and trigger an AR label to be displayed, or Text Recognition and Translation to detect, read, and translate text in an image. Augmented reality APIs are then used to overlay translated text back into the 3D world.



Autonomous Vehicles

One of the biggest adopters of CV technology are automotive companies striving to make autonomous vehicles. This use case highlights so many types of image and video annotations working together. For example, using Video Object and Event Tracking to understand how objects move through time. This is useful for tracking pedestrians, cars, and cyclists as they enter and exit the area of interest over many frames of videos and LiDAR scenes. Point Cloud Labeling (LiDAR, Radar data) would also be used to understand the scene in front of and around the car by identifying and tracking the objects in the scene. Point cloud and video data could be merged into one scene to be annotated, allowing CV models to fully understand the world and environment around the vehicle.

Healthcare

CV and healthcare are a natural fit due to the availability of medical imaging data. Doctors at many hospitals use CV to support and supplement their diagnoses. Algorithms analyze images from MRIs, X-rays, and CT scans to identify signs of infection or disease. In many cases, deep learning neural networks are trained to extract features from medical images to learn how to identify anomalies. For more precise image analysis, teams require pixel-level semantic segmentation to identify tiny polyps or tumors for certain types of cancer. In both scenarios, CV has potentially life-saving implications.

A lesser known use case in healthcare is also early detection of autism in children. CV-based AI can apply gaze tracking algorithms to identify unusual patterns early on, which could indicate cognitive impairments.

Retail

CV systems track customer behavior in stores to provide retailers important insights on customer needs. For example, cameras can track movement patterns, detecting which areas of the store experience more frequent traffic. The AI can also track customer gaze, seeing which displays attract the most attention. Using these behavioral analytics, retailers can set up their store layouts to drive the most sales. Retailers can also use CV with facial recognition to automatically spot suspicious behavior or known shoplifters, leading to improved loss prevention.

A few major retailers have also started using robots to track store inventory. These robots capture images of store shelves, which the algorithm analyzes to determine which products are low-stock. Robots can also use image transcription to scan barcodes for product information.

Manufacturing

Predictive maintenance is just one way CV benefits manufacturers. Cameras and sensors installed on equipment monitor for potential maintenance needs, and a CV algorithm can indicate when those needs must be met prior to any downtime or significant repairs.

CV systems are also used to monitor product quality, quickly detecting defects in components. Manufacturers can then take swift action to manage the issue accordingly.

Agriculture

In recent years, the farming industry has become increasingly tech-savvy. There are several applications of CV in agriculture: using drone imaging and pixel-level semantic segmentation on images of crops and weeds, farmers can pinpoint the exact areas of the field to spray fertilizer versus pesticides, decreasing waste. Farmers also use object detection to predict crop diseases or pest infestations, allowing them to manage the situation before it becomes too late.

Farmers also use CV algorithms to sort crops by size, weight, or quality, enabling more efficient distribution.

Additional Use Cases of Computer Vision

- 3D model building (photogrammetry)
- Automotive safety
- CGI in movies
- Fingerprint recognition and biometrics
- Face recognition
- Sign language recognition and translation
- Scouting potentially hazardous terrain
- Human pose estimation for sports

Challenges in the Field of Computer Vision

Launching a successful Computer Vision project isn't without its challenges and the field itself is still working to answer questions around how CV should work. Before endeavoring to deploy CV-based AI, it helps to have knowledge on key areas where progress still needs to be made.

Understanding Human Vision

To mimic the perceptive qualities of the human eye and brain isn't an easy feat. We still don't have a full understanding of how human vision works—how its biological parts function and how the brain interprets what we see. We also have life experiences that help us add context to what we see. Once we know more in the area of human vision, we can better replicate its functions and interactions in deep learning algorithms for CV.

Complexity of Visuals

We can see and identify objects from various orientations, lighting conditions, and obstructions. Programming or teaching a CV system to do the same is a monumental effort, requiring millions of examples. Prior to beginning a CV project, teams must make decisions on how they'll handle these less-than-ideal conditions.

Label Consistency

Determining classes and labels, and differentiating them enough from each other to be meaningful, is an important step in a CV annotation project. If the classes are too similar, this can be confusing for annotators and algorithms. Alternatively, if the classes are too vague, you're not extracting the full potential information from your data.

Data Collection

If one thing should be clear, it's that data plays a critical role in CV. That said, despite the growing availability of data, it still remains challenging to source enough high-quality data to support a CV project. Even if you're able to do so, transforming raw data into structured data is a tremendous effort on the part of the organization and its annotators.

Turning to third-parties for help can be a strategic step in getting your Computer Vision project off the ground.

Many companies offer 'off-the-shelf' datasets which is pre-annotated data, available immediately, making them a great option to assist in building quality training data for your AI/ML models.

Considerations for Success

Computer Vision is a prime area of opportunity for organizations looking to get into the AI game. If you're thinking about launching a CV project, there are several best practices to consider before getting started:

Define Scope

Begin your project with a clear and narrow definition of your business goals. Your team members should all understand these goals and how their role supports achievement. Be sure that a CV-based approach is the most appropriate solution to the problem you're trying to solve. The requirements of your project (especially in terms of data and data processes) should stem from your business goals.

Determine Success Criteria

Decide on clear mathematical metrics that will indicate success of your project. Include metrics that track data bias as well; these are often missed in planning. Track these metrics throughout model build and post-deployment using KPI dashboards.

Integrate Resources

Leverage both human and machine to drive speedy, efficient deployments. You may want to incorporate multiple algorithms and processes to support your project, as these can provide essential accuracy checks on each other. Redundancy, especially during annotation, drives greater confidence in your results and reduces risk of failure.

Build to Scale

Construct a scalable, automated training data pipeline for two reasons: one, you can use this pipeline to continuously retrain your model, and two, you can replicate this pipeline in future projects.

Plan to Iterate

The process of building out a CV model is one of frequent iteration. Your initial requirements will likely change throughout the project as you discover more edge cases that need to be accounted for and work toward higher-performing models.

Creating real-world AI solutions may be a significant undertaking, but empowering humans with the right technology has always led to an enhancement of human effort. By collaborating on knowledge sharing and best practices, we can all hope to continue the dialogue on how AI should be used fairly and for advancement in our world.

Smart Labeling

Machine learning assistance to accelerate ROI on your AI initiatives

At Appen, we provide high-quality annotated training data to power the world's most innovative machine learning and business solutions. We help to identify and process objects in images and videos and the data is used to build intelligent autonomous systems used for self-driving cars, robotics, mapping & satellites, agriculture technology and more. Many of our annotation tools feature Smart Labeling capabilities, which leverage machine learning models to automate labeling and enable contributors to work quickly and more accurately.

Pre-Labeling

Machine Learning provides an initial 'best guess' hypothesis before contributors start the task. With human contributors reviewing pre-processed annotations instead of starting a judgment from scratch, the time needed to annotate data drastically reduces.

Speed Labeling

Machine Learning provides for in-tool efficiency, quality and accuracy, improving ergonomic conditions while contributors work. This reduces cognitive strain and allows contributors to work faster and more comfortably, increasing the throughput of their annotations.

Smart Validators

Machine Learning verifies human judgments before they are analyzed. This ensures you are only paying for quality, eliminating the need for peer reviews and the risk you're paying for judgments that don't meet your requirements.

With Appen's platform, improve quality and shorten annotation time for your Computer Vision projects so you can build and deploy AI faster and with more confidence.

Learn more about what annotation capabilities we have available to help you with your video annotation projects, or contact us today to speak with someone directly.

About Appen

Appen collects and labels images, text, speech, audio, video, and other data used to build and continuously improve the world's most innovative artificial intelligence systems. Our expertise includes having a global crowd of over one million skilled contractors who speak over 235 languages, and the industry's most advanced AI-assisted data annotation platform. Our high-quality training data gives leaders in technology, automotive, financial services, retail, healthcare, and governments the confidence to deploy world-class AI products. Founded in 1996, Appen has customers and offices globally.

- Experience working in **170+ countries**
- Expertise in **235+ languages**
- Over **1,125 employees** located in offices around the globe
- Access to a curated crowd of over **1 million** flexible contractors worldwide
- Nearly **1 billion** judgments made and **3 million** images and videos collected **in 2020**
- **25 years working** with leading global technology companies

appen