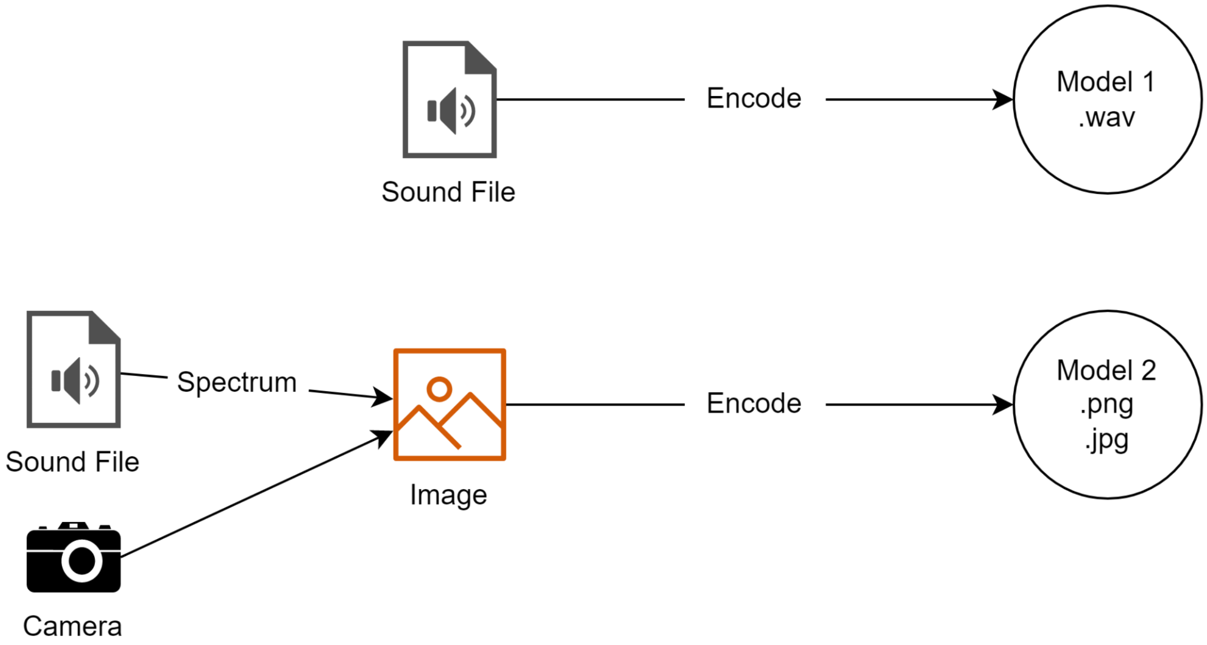
AI Write-up

Image Classification



Demonstration of different ways AI models can accept sound input. Model 1 accepts an encoded sound file, and Model 2 accepts an encoded image, either from a camera or a spectrogram.

Figure 1

Source: https://levity.ai/blog/image-classification-in-ai-how-it-works#:~:text=Image%20classification%20analyzes%20photos%20with,to%20improve%20your%20business%20operations

Image classification involves identifying and labeling objects in an image based on certain criteria. This can be done with single-label classification, where each image has only one label, or multi-label classification, where images can have multiple labels. The steps for image classification are listed below:

1. Pre-processing: Preparing data by cleaning it before giving it to the AI model. This includes removing duplicates, cutting irrelevant data and outliers, and detecting missing data.
2. Object detection: Locating an object in an image. For example, in autonomous vehicles, this could be locating another vehicle.
3. Object recognition and training: Assigning labels to located objects and then using labeled data to train the AI model.
4. Object classification: The AI model compares an image's object patterns to the patterns that it was trained on, and then classifies these objects.
5. Connecting to an AI workflow: Define where new data comes from and what is output once new data has been classified.

Example AI image classification models we used for practice:

- https://keras.io/examples/vision/image_classification_from_scratch/
  This example classifies animals in images as either cats or dogs.
- https://www.tensorflow.org/tutorials/keras/classification
  This example classifies 10 different types of clothing.

Image Tagging

Image tagging is automatically assigning descriptive labels to an image or group of images. The goal is to make finding and organizing images faster, easier, and more efficient. This process involves using algorithms to identify and classify objects within an image. Image tagging techniques can be used for various applications such as content-based image retrieval (CBIR), computer vision, object recognition, facial recognition, and many more.

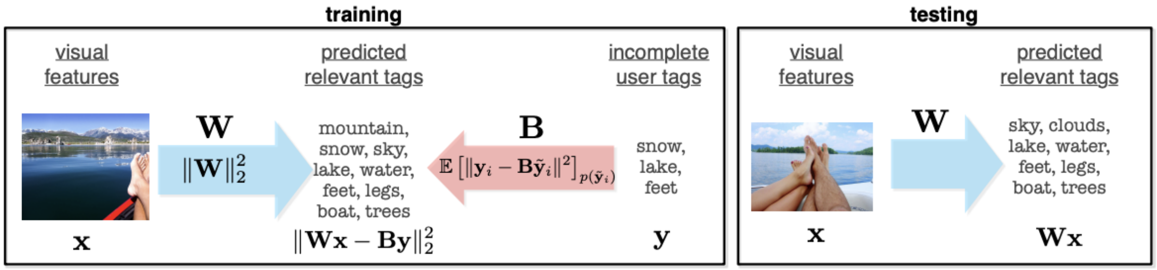

training
visual features
predicted relevant tags
incomplete user tags
$\mathbf{W}$
$\|\mathbf{W}\|_2^2$
mountain, snow, sky, lake, water, feet, legs, boat, trees
$\mathbf{B}$
$\mathbb{E}\left[\|\mathbf{y}_i - \mathbf{B}\tilde{\mathbf{y}}_i\|^2\right]_{p(\tilde{\mathbf{y}}_i)}$
snow, lake, feet
$\mathbf{x}$
$\|\mathbf{Wx} - \mathbf{By}\|_2^2$
$\mathbf{y}$
testing
visual features
predicted relevant tags
$\mathbf{W}$
sky, clouds, lake, water, feet, legs, boat, trees
$\mathbf{x}$
$\mathbf{Wx}$

Figure 2

Image tagging consists of two steps: feature extraction and classification. During feature extraction, the algorithm analyzes the pixels in an image to determine various characteristics like color, texture, shape, size, etc., which are then used as features that the algorithm can use for classification. Once these features are extracted from the image data set, a classifier algorithm is trained on labeled training data to learn how to recognize certain objects based on their features. For example, if an algorithm is trained on labeled images of cats and dogs with specific features extracted from each image (e.g., fur color/length/pattern), it will be able to recognize these same cats and dogs when presented with new images with similar characteristics.

The level of accuracy obtained during this process depends on several factors, such as the quality of the training data set used for training the classifier model, the number and type of features extracted from each image, the complexity of the problem being solved by the model (i.e., number of classes being identified) and the type of classifier model being used (i.e., shallow versus deep learning models). In addition to accuracy offers efficiency gains since fewer resources are needed for a large volume of tag requests than manual labor or human experts.

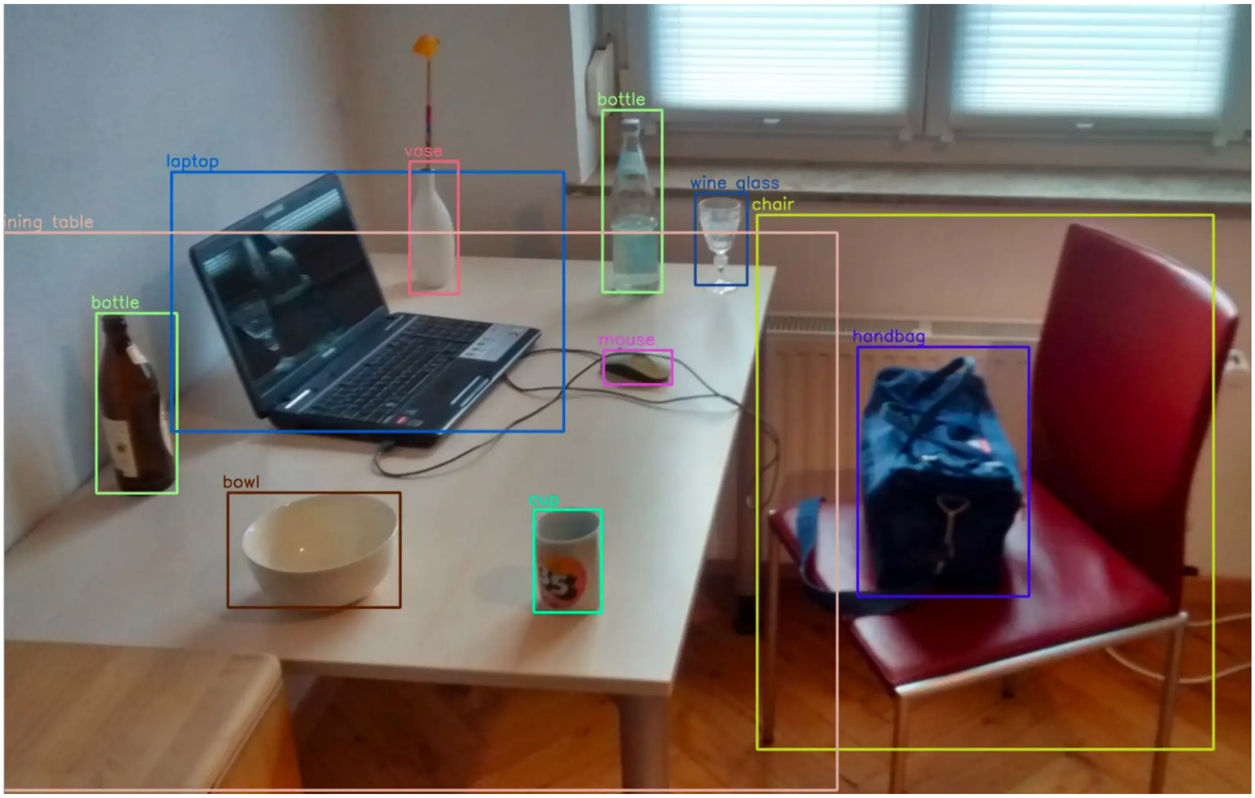


Figure 3

Given its potential for increasing speed and accuracy while simultaneously lowering costs in tasks such as content-based image retrieval (CBIR), automated image tagging has become increasingly popular in recent years within businesses that deal with large volumes of digital assets such as photos or videos. Automated tagging enables these organizations to effectively organize their assets according to their needs or wants quickly and efficiently without manually tagging each assert individually - something that would be incredibly time-consuming and costly if done manually instead.

Images source:

Figure 2
https://www.microsoft.com/en-us/research/wp-content/uploads/2013/01/ICML13-final.pdf
Figure 3
https://datagen.tech/guides/image-annotation/image-labeling/

Metadata

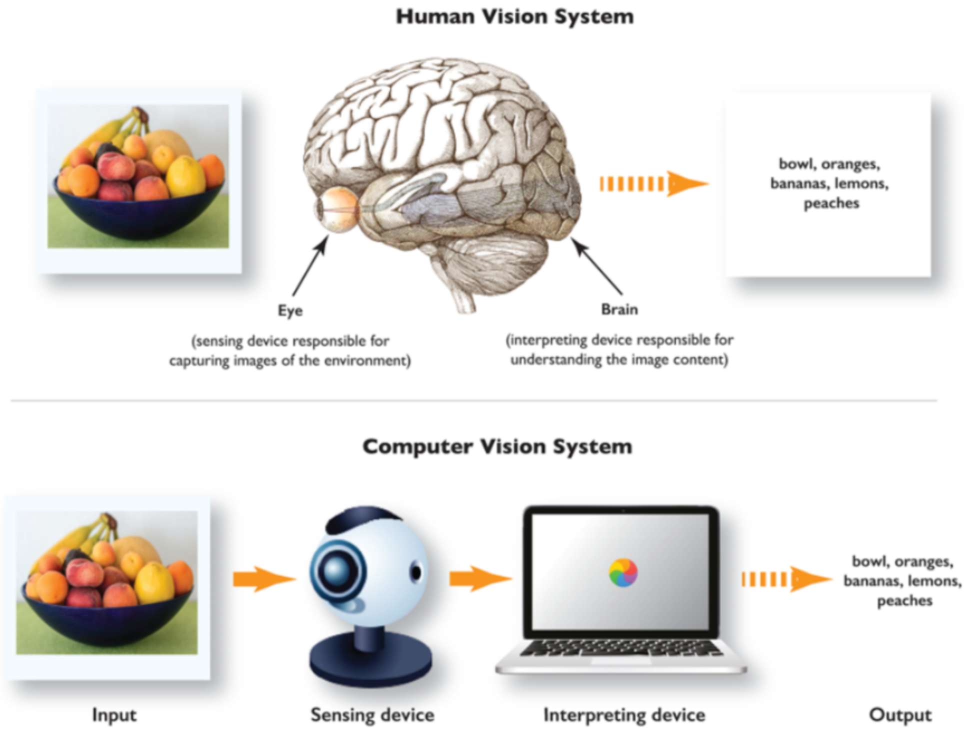
The internet has changed how we use metadata, which is now more important than ever. Tags are used to describe and organize data, making it easier to retrieve relevant information. They can be applied to web pages, images, and documents, allowing users to quickly locate the information they need. Tags have been around since the dawn of the internet, but their use has become much more sophisticated with the advent of machine learning and artificial intelligence (AI).

Machine learning algorithms enable computers and other devices to learn from experience without being explicitly programmed. Computers can then identify patterns in large data sets using clustering, classification, or regression analysis techniques. This enables them to recognize trends and predict future events or outcomes. AI allows machines to understand natural language by mapping words and phrases to various concepts and knowledge bases. AI systems can process natural language queries and respond intelligently with relevant results using search engines like Google or Bing.

Tags are combined with machine learning algorithms to rapidly identify page content or document content. For example, when searching for a specific topic on Google, tags associated with related topics will appear as suggestions alongside your query term. Machine learning-powered recommendation systems also use tags along with user behavior data to recommend webpages or products you might find interesting based on what you have previously searched for or bought.

In addition to helping users quickly access content or products they need, tags combined with machine learning algorithms can also be used for personalization purposes. By analyzing user behavior across multiple sources, such as web searches, online transactions, and social media interactions, marketers can tailor advertisements specifically for individual customers' interests by targeting keywords most likely relevant to them.

Computer Vision


Human Vision System
bowl, oranges, bananas, lemons, peaches
Eye
(sensing device responsible for capturing images of the environment)
Brain
(interpreting device responsible for understanding the image content)
Computer Vision System
bowl, oranges, bananas, lemons, peaches
Input
Sensing device
Interpreting device
Output

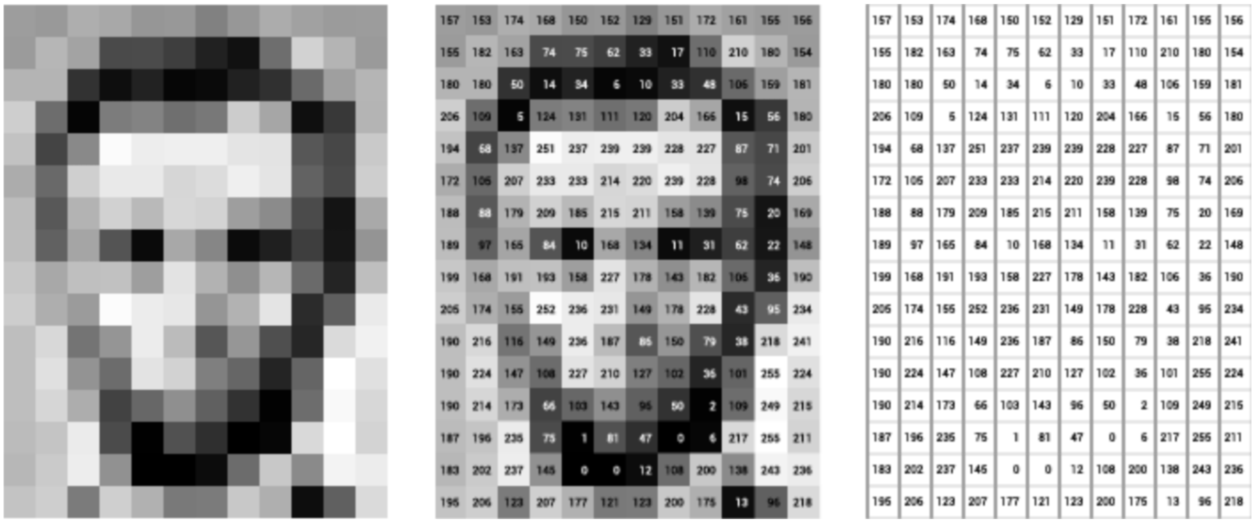
Human Vision System vs. Computer Vision system

Figure 4

Computer vision has become an increasingly important part of machine learning. It is a branch of artificial intelligence focused on helping computers understand and interpret digital images or videos, much like humans do. Computer vision also helps machines recognize objects, identify trends, and make real-time decisions. This technology has enabled medical image analysis, autonomous vehicles, facial recognition systems, and more breakthroughs.

Object recognition is one of the most common applications of computer vision in machine learning. By using algorithms to analyze digital images or videos, machines can recognize objects such as faces, animals, plants, and cars with remarkable accuracy. This technology has been used to power self-driving cars, face recognition systems for security purposes, and image-based search engines, which identify similar items in databases.

Another computer vision application in machine learning is optical character recognition (OCR). OCR enables machines to read text from printed documents or even handwriting on paper and digital documents. This technology has been used by companies such as Google Drive and Tesseract to convert scanned documents into editable text formats such as Word files or PDFs.



157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

Machines interpret images as a series of pixels, each with their own set of color values

Figure 5

Computer vision also plays a role in facial recognition systems which are becoming increasingly popular due to their ability to accurately detect faces from large databases quickly and accurately. Facial recognition systems use deep learning algorithms to analyze photographs or videos for facial features such as eyes, nose, mouth, and ears, which are then compared against a database of known faces to determine identity.

In medical image analysis applications, computer vision is used for automated tumor detection from MRI scans with remarkable accuracy. In this case, the AI system processes the MRI scan images using algorithms that can detect small tumors that might otherwise be undetectable by human eyes or traditional medical imaging techniques such as CT scans or X-rays.

Images source:

Figure 4
https://www.v7labs.com/blog/what-is-computer-vision

Figure 5
https://www.datarobot.com/blog/introduction-to-computer-vision-what-it-is-and-how-it-works/

Face Recognition

Face Recognition is an important process in machine learning that involves identifying and verifying a person from their face. It can be used for various applications, such as automated passport control, security systems, and access control. The process of face recognition involves comparing two facial images and determining if they are a match or not. This done by extracting feature points from the faces and then using those feature points to create a mathematical model. These models are then compared to the existing database to determine the individual identity in the image.

The process typically begins with preprocessing, where noise is removed from the image, edges are enhanced, brightness is adjusted, and other modifications are made to improve quality. After this, feature extraction occurs where certain features like eyes, nose, and mouth are detected to generate a numerical representation of the facial characteristics. These representations are often referred to as face embeddings which can be used for comparison with other faces in the database.

A similarity measure is calculated to compare two images, comparing both embeddings against one another. If the results indicate a high similarity between both embeddings, then it can be said that the two individuals have similar facial structures. On the other hand, if there is a lower similarity between them, then it means that they have different facial structures and hence should not be matched together. In addition to this basic comparison technique, many advanced algorithms, such as deep learning/deep neural networks, have been developed over recent years, which allow more accurate matching than traditional methods.

Other improvements, such as 3D facial scans, can also be used for better accuracy in facial recognition tasks due to their increased ability to capture subtle differences between facial features. Additionally, techniques such as adaptive thresholding or voting mechanisms can also be employed, which take into account multiple samples before deciding whether two images belong to the same or different people. Such techniques help increase accuracy while keeping the false positive rate low during facial recognition tasks.

Sound

Speech-to-Text Recognition (STR)

Speech-to-text recognition is the ability of a machine or computer to recognize words spoken aloud and be able to translate them into visual text. Speech-to-text recognition is quite similar to voice recognition but in its case, it is mainly focused on the what and not the who. That is to say, speech-to-text recognition is mainly focused on what is said while voice recognition is mainly focused on who is speaking. Speech-to- text recognition can be classified under Natural Language Processing which is a branch that focuses on training computers to understand human speech and text.

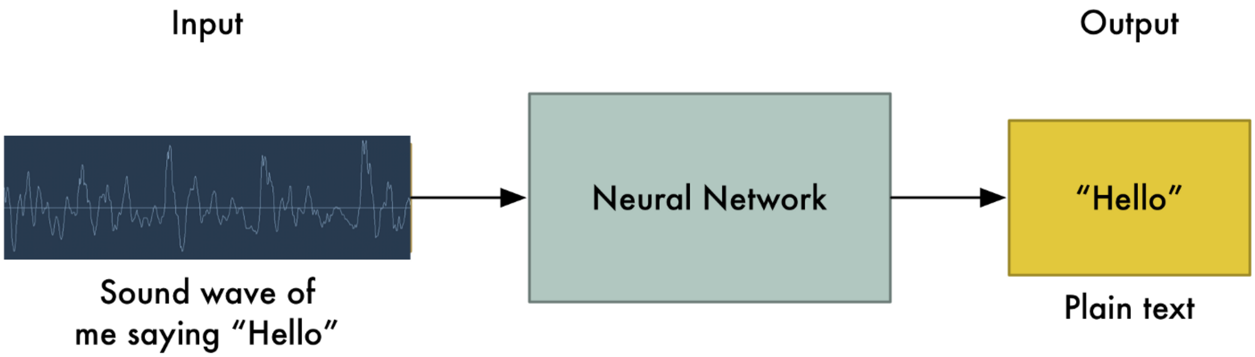


Figure 6

Types of Speech To Text

- Speaker Dependent: Dictation software
- Speaker Independent: Phone applications

Applications

- Call analysis
- Media content search
- Media subtitling
- Clinical documentation

Limitations

Although it might sound like a game changer nothing is ever perfect, STR sometimes is inaccurate and would require human inputs and edits for optimal usage. It also requires that the audio input be it prerecorded or live, must be as clear as possible in order for the model to recognize whatever is said.

Sound Patterns

By analyzing a sound's amplitude and frequency, an AI model can find patterns and categorize sounds based on these patterns. The sound can be analyzed directly or it can first be converted into a spectrogram, which is an image showing changes in a sound's frequency over time. Once an AI model is given a spectrogram, it can treat it like any other image and the sound patterns problem becomes an image classification problem. The following is an AI model using this process that we practiced with:

- https://www.tensorflow.org/tutorials/audio/simple_audio
  AI model that analyzes spectrograms of spoken keywords.

Encoding

Sources:
https://towardsdatascience.com/6-ways-to-encode-features-for-machine-learning-algorithms-21593f6238b0
https://ai-ml-analytics.com/encoding/

In order for an AI model to understand the data it receives, it must be encoded as a number. Encoding converts a categorical variable into a numeric variable. There are two types of categorical variables:

1. Nominal categorical variables are those for which arrangement does not matter. An example of this would be a variable for the class of an animal which could be cat or dog. The two categories would have no rank.
2. Ordinal categorical variables are variables where the order is important. They can be ranked based on criteria. An example of this would be a variable for size (small, medium, or large). The categories could be ranked from smallest to largest or vice versa.

The following are six ways to encode categorical variables:

1. One-hot/dummy encoding (for nominal variables): Represent data as a combination of zeroes and ones. Use a separate dummy variable for each category and set the value of the dummy variable to 1 if the data belongs to that category and 0 otherwise. Problematic for categorical variables with large amounts of categories because it results in large numbers after encoding.
2. Label/ordinal encoding (for ordinal variables): Assign each category with a label (number) based on rank. For example, 1 for small, 2 for medium, and 3 for large.
3. Target encoding: Replace the category with the mean of the target values. This ranks categories by the effect they have on the target.

4. Frequency/count encoding: Count encoding replaces categories with their counts in the data. Frequency encoding does the same thing but normalizes the counts in order to reduce the effect of outliers.
5. Binary encoding: Encode categories as integers and then convert them into binary numbers.
6. Feature hashing: Use a hashing function to encode variables. Good for variables with large amounts of categories because it is fast to compute and uses a fixed-size array that does not grow in size when new categories are added.

Convolutional Neural Network (CNN)

Source:
https://www.mathworks.com/discovery/convolutional-neural-network-matlab.html

Convolutional neural networks (cnn or convNet) are deep learning network architectures that learn directly from data. CNNs are mostly used to recognize patterns in pictures to identify objects and classes; they are also used to classify audio. A CNN is made up of numerous hidden layers between the input and output layers. These layers apply operations to the data to change it while learning features particular to the data. The three most popular layers are pooling, activation (ReLU), and convolution.

Convolution: puts the input images through a set of convolutional filters, each of which activates certain features from the images.
The Rectified Linear Unit: is the most commonly used activation function in deep learning models. The function returns 0 if it receives any negative input, but for any positive value x, it returns that value back. So it can be written as f(x)= max(0,x).
Pooling: simplifies the output by performing nonlinear downsampling, reducing the number of parameters that the network needs to learn.

These operations are repeated over tens or hundreds of layers, with each layer learning to identify different features.