OPTICAL CHARACTER RECOGNITION — A Complete Guide

Zobia Khan, Muhammad Saad, Wajahat Sarwat, Muhammad Anas
Pakistan, Karachi
NED University of Engineering and Technology

Abstract—**OCR or optical character recognition is the technology used to distinguish printed or handwritten text within digital images of physical documents. OCR is most commonly used to turn hard copy, legal or historic documents into PDF. An OCR system analyzes an image and identifies dark areas as characters that need to be recognized, enabling digital archiving, editing and documents searching with common programs like Microsoft Word or Google Docs. Characters are recognized with one of two algorithms. Pattern Recognition: when OCR programs are fed examples of texts and various fonts and formats to compare and recognize characters in a document, or Feature Detection: when OCR programs apply rules about letter and number of features, like lines, curves angle in order to recognize characters. For instance, the letter A might be recognized as two diagonal Lines connected by horizontal line across the middle. OCR is also used for archiving newspapers and phone books, mobile App check deposits, automated tollbooth collection, indexing print materials for search engines, managing legal documents, sorting mail and digitizing documents to be read aloud for the visually impaired. Prior to OCR technology, paper documents could only be converted manually with someone typing documents one by one. OCR saves a huge amount of time, reduces translation errors and minimizes effort.**

1. Introduction

In computer sciences, artificial intelligence, sometimes also called machine intelligence, is intelligence demonstrated by machines, contrary to natural intelligence displayed by human beings.

John McCarthy, father of AI, defines artificial intelligence as:

”The science and engineering of making intelligent machines, especially intelligent computer programs.”

According to MIT professor Patrick Winston:

”It’s about algorithms enabled by constraints exposed by representations that model targeted thinking, perception and action”

AI is basically a computer program. How we human beings can learn by observing things, similarly machines too learn from surroundings and are able to think on their own.

The difference between a traditional and AI based program is that on a traditional programming approach, the input gives us the desired output but in an AI based approach, when the program does things repeatedly, it will think and do that work in a better way, in less time.

2. Machine Learning

Machine learning is simply how computers “think” through and execute a task without being programmed to. It is a subset of artificial intelligence that involves algorithms and models that can automatically analyze and learn data to make inferences and decisions without human intervention.

Tom Michael Mitchell, American scientist and author of book Machine Learning, described the machine learning process as,

“A computer program is said to learn from experience E in respect to some class of tasks T and performance P if its performance at tasks in T, as measured by P, improves with experience E.”

In simple terms, machine learning describes how computers perform tasks on their own by learning from past experiences. The process of learning from experience and performing tasks uses a series of instructions called algorithms, which make up computer ”ideas.” Machine learning falls into two categories:

1) Supervised machine learning
2) Unsupervised machine learning

2.1. Supervised Machine Learning Algorithms

In supervised machine learning, you train the system on a data set of categorized examples, which the system can rely on to make conclusions or predictions. These labeled examples are already highlighted with your correct answers to help the system make the correct correlations. After sufficient training with the training data set, the system can make accurate predictions about the result.

For example, if the system or device is to help you predict how long it will take to drive from your home to your workplace, it should be trained using data that contains the time spent driving to work from home in different weather conditions, throughout different routes and sometimes. Different days and days of the week. Using this training data, the machine can deduce which routes take the longest to get to work, which weather conditions prolong your commute, and what time of day driving to work will be fastest.

This data set forms a series of ”insights” with which the machine can tell you how long it will take to drive to work on any given day.

2.2. Unsupervised Machine Learning Algorithms

Unsupervised machine learning algorithms train a system on unclassified or dis-aggregated data. So in this form of machine learning training the system does not receive correct responses and therefore does not need to present an accurate output value. Instead, you are able to draw conclusions that describe hidden information from unlabeled data

Unsupervised machine learning algorithms are used primarily in image recognition applications. For example, you can create an automated form that can identify people laughing in a video without actively training you to recognize them. The machine deduces similar patterns from people laughing and associates these patterns with text, voice and speech in the video.

Although the model is not told whether such conclusions are true or false, unlike supervised learning, the machine builds confidence and solidifies these conclusions upon subsequent exposure to such patterns.

This form of machine learning reflects unsupervised human learning behavior, such as visual recognition. For example, a child sees her father's car and identifies it as a car. A few days later, he saw a neighbor's car and quickly concluded that it was a car, without telling him, noticing similar patterns: shape, features, and sound.

2.3. Semi-supervised machine learning algorithms

Somewhere between supervised and unsupervised algorithms are semi-supervised algorithms, which use classified and unclassified data to train machine models.

3. OCRs - Optical Character Readers

OCR is a technology that converts printed text into a digital format that is view able and editable. The variety of fonts and forms of writing a single character makes this problem difficult to solve.

So far, OCR technology has advanced somewhat, but a 100% OCR job is still difficult to achieve, that is, due to the resolution of the file or when the text in the original file is clear enough to recognize.

AI-based OCR engines combine artificial intelligence and OCR technology to transform a document into an editable and machine-readable digital format. Traditional OCR engines are supposed to analyze a document in image form based on patterns when the image contains text and then extract the text into a machine-readable format. This helps to convert scanned documents to a digitally editable format while comparing available character images with those in your database for traditional OCR engines.

AI-based OCR engines use various machine learning, computer vision, and natural language processing (NLP) algorithms to create text and images and deliver more accurate results to users. By observing and understanding the language, document type, context, and other specific details of the document, AI OCR engines build a comprehensive understanding of the document and the data it contains. With an accuracy of up to 99.9%, AI OCR engines eliminate the need for a human resource to make corrections.

Some of the more popular OCR programs include:

1) Capture2Text
2) Google Keep
3) OneNote
4) PhotoScan
5) SimpleOCR
6) Tesseract

Before an OCR algorithm can be selected, the image must undergo preliminary processing. In this step, the document is straightened, speckled, and converted from color to a binary image, an image where the only two colors are black and white. The feature detection algorithm identifies a character by analyzing the lines and lines that compose it.

The second approach, pattern recognition, works by identifying the character as a whole. We can identify a line of text by looking for rows of white pixels with rows of black pixels in between. In the same way, we can identify where an individual character begins and ends.

Next, we convert the character image to a binary matrix where the white pixels are 0 and the black pixels are 1. Using the distance formula, we can calculate the distance from the center of the matrix to the furthest 1. Then we create a circle of that radius and divide it into more curved sections. At this point, the algorithm compares each subdivision against a database of arrays representing characters with different scripts to find the character with which it is statistically most common. This for each line and each character facilitates the incorporation of print media into the digital world.

3.0.1. OCR Common Processes.

- **Scanning:** Scanning image is the first step of OCR and also an important one. The accuracy and detection depends on this phase. The image should be clear and well preserved and in good pixel quality.
- **Pre-Processing:** In this step the image is prepared for OCR process. We increase the contrast, brightness, noise and also simplify the colors.
- **Processing OCR:** A crucial step in which the desired OCR model system reads the image and extracts out the data. It is necessary to choose a model that fits the desired problem.
- **Post-Processing:** In this the final output is displayed and corrected to the nearest phase if not detected properly.

4. OCR Use Cases

OCR is an emerging technology to convert previously-paper based data to digital data.

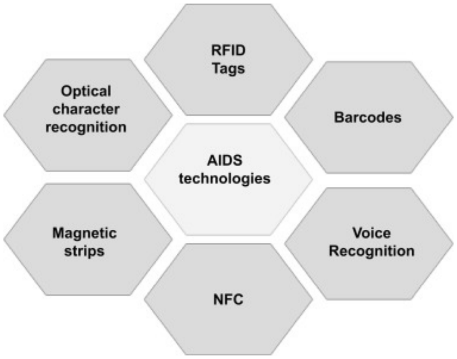

Figure 1. OCRs in different forms

- Handwriting detection (such as OCR)
- Image recognition and video recognition.
- In political science, a potential voter is classified as ”will vote” or ”will not vote”, or as a ”Democrat” or ”Republican voter.”
- Credit Scores - Collect financial characteristics by comparing them to people with similar financial characteristics to those in a database.

The different forms of OCRs are being used in almost every industry. The application was however restricted to some particular applications before.

The key modules that make up an end-to-end system for handprint OCR applications are preprocessing (form identification, field and line isolation), character segmentation, segment reconstruction, character recognition, and word and phrase construction without human intervention or human correction. For many of these modules, methods based on two primary approaches have been taken: rule-based and image-based recognition[1].

4.0.1. OCRs are no more used stand-alone. OCRs are no longer used stand-alone because the images captured from it are either skewed or in unorganized format.Backgrounds with colors or textures are hindrance to get accurate result and only high quality images lead to better OCR accuracy.

4.1. KNN model approach

The K-Nearest neighbor algorithm is used in the cluster.

Clustering is the practice of taking data points on a graph (much easier in 1D or 2D) and logically assigned groups. In this graph we can see three quite clear groups. Everyone is assigned to a group. For example, it could be three different groups in a drug trial.

AI is not good at recognizing clusters as we are, so one of the algorithms we use to help them is K-Next neighbors.When we add a new point to the graph, we look at the nearest neighbor of K, where K is a few (odd, to avoid drawings). Each of these points should already have a classification (from previous uses), and we will assign the new point to the group that has the most of its neighbors.

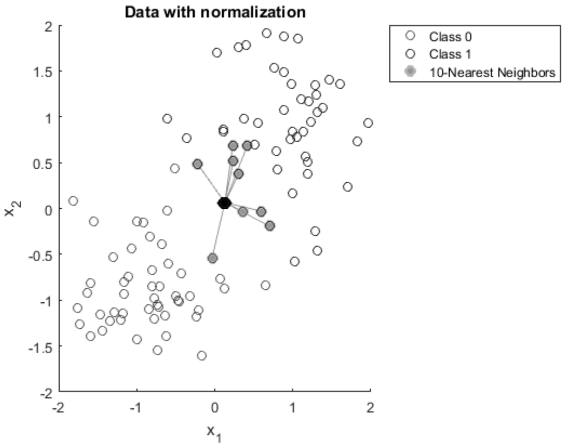

Figure 2. KNN Algorithm Representation [2]

W.e.g. If I specify that a point is around (-4,6) (in the upper right connection), then the closest neighbors are all in that star cluster (say A), so let’s say it belongs to A.

However, if you were to assign a point to be at (1.5), it would be much closer and at a glance, we cannot tell that it is obviously part of a star cluster.

We can change this by the weight of the KNN distance (fairly self-explanatory), and the algorithm named K is similar (the center of each star cluster moves according to the mean values of things in its star cluster. stars it owns is determined by KNN).

k- The nearest neighborhood algorithm (kNN) is one of the simplest non-parametric lazy classification algorithms. Its purpose is to use a database in which data points are separated into different classes to predict the classification of a new sample point.

Nonparametric means that no preconditions are required for the distribution of basic data, this characteristic makes it a bit significant. Therefore, KNN can and could be one of the first options for a classification study if there is little or no knowledge of the distribution data.

Another key feature of kNN is that there is no explicit training phase. This also means that the training phase is relatively quick. KNN can also be used for regression forecasting problems. Therefore, it has predictive power.

4.1.1. Implementation using KNN Algorithm. We can implement a KNN model by following these steps:

1) Upload the data.
2) Initializes the value of k.

To get the predicted class, iterate from 1 to the total number of training data points. Calculate the distance between the test data and each line of training data. Here we will use the Euclidean distance as the distance metric because it is the most popular method. The other metrics that can be used are Chebyshev, cosine, etc. Order the calculated distances in ascending order based on the distance values. Get the first K rows of the ordered matrix. Get the most common class of these series. Return to the scheduled class.

4.2. KNN vx SVM vx Others and Why KNN?

When choosing an algorithm, it depends on several factors. KNN tends to work well when you have many instances (points) and few dimensions, but you need to be very careful about performance becaus a brute force version of KNN can be very slow when you have a lot of data. So KNN is good if you have a lot of points but at the same time it gets too slow in a catch-22 scenario.

KNN will beat a linear SVM if the data is nonlinear (doh) but that is not true for kernel based SVM. A very good question would be to find out if there is a case where KNN is better than RBF SVM, I'd rather say no.

KNN is also very sensitive to bad characteristics (attributes), so the selection of characteristics is also important. KNN is also sensitive to aliens and removing before using KNN tends to improve results.

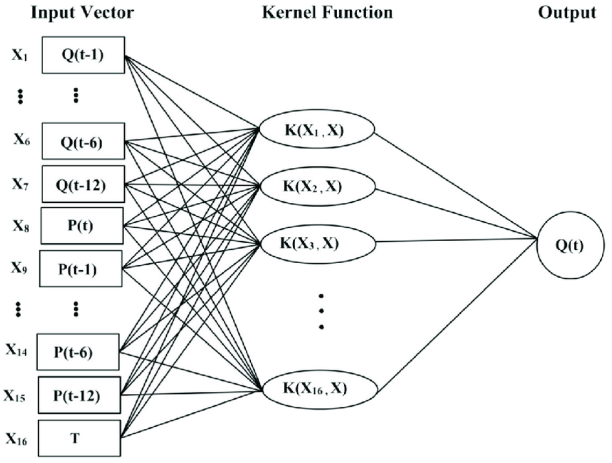

Figure 3. General Structure of SVM model

SVM can be used in a linear or non-linear way with the use of a kernel, if you have a limited point in many dimensions, SVM tends to be very good because it could find the linear separation that should exist. SVM is good with aliens because it only uses the relevant points to find a linear separation (support vectors)

SVM needs to be tuned, the ”C” cost and the use of a kernel and its parameters are critical hyperparameters for the algorithm.

So make something useful out of this mess:

- If you have many points in a low-dimensional space, KNN is probably a good choice.
- if you have a few points in a high-dimensional space, a linear SVM is probably better.

4.3. Why Training Model on GPU is necessary?

GPUs are a bunch of simpler core processor than complex CPUs. However GPUs arnt fast than CPU but GPUs are designed to perform certain tasks fast when given a large dataset.The main reason why GPU work well when training model is because they work on SIMD (Single Instruction, Multiple Data).

4.3.1. *What do AI have to do with Graphics?*.

AI is a heavy task. A single NVIDIA graphics card equals the power of Kasparov's vanquisher. Meaning, a single GPU chip can can process a thousand times more dataset than our typical x86 processors.

To train our model on GPU, we need to check if we have CUDA enabled GPU with Compute Capability 3.0 or higher and install GPU supported version of Tensorflow. CUDA is a platform developed by NVIDIA that allows you to use their GPUs for general computing. Installing CUDA is necessary to run popular ML frameworks, such as Pytorch and Tensorflow, on NVIDIA's GPUs.**cuDNN-** to install tensorflow-gpu, you will need to install cuDNN, which is NVIDIA's GPU-accelerated deep neural network library.[3]

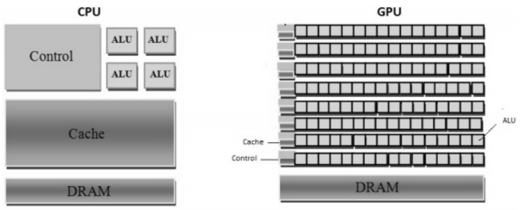

Figure 4. Fundamental design of CPU vs GPU

Implementation of kNN on GPU is an ongoing research from last few years, focusing on improving the performance of kNN. By considering these aspects, our research has been started and found a gap in this research area. This master thesis shows effective and efficient parallelism on multi-core of CPU and GPU to compare the performance with single core CPU.[4]

4.4. OCRs for Text Translation

To further advance OCRs they can be used for the translation of sentences to desired languages. An RNN algorithm works the best for building OCRs at small level.The complexity of the problem is determined by the complexity of the vocabulary. A more complex vocabulary is a more complex problem. A suggested apporoach would be:

4.4.1. *Pre-Process:*.

We convert the text into sequences of integers using the following preprocess methods:

1) Tokenize the words into ids
2) Add padding to make all the sequences the same length.

4.4.2. *Tokenize:*.

For a neural network to predict on text data, it first has to be turned into data it can understand. Text data like ”dog” is a sequence of ASCII character encoding. Since a neural network is a series of multiplication and addition operations, the input data needs to be number/s.

We can turn each character into a number or each word into a number. These are called character and word ids, respectively. Character ids are used for character level models that generate text predictions for each character. A word level model uses word ids that generate text predictions for each word. Word level models tend to learn better, since they are lower in complexity, so we'll use those.

4.4.3. ***Padding:.*** When batching the sequence of word ids together, each sequence needs to be the same length. Since sentences are dynamic in length, we can add padding to the end of the sequences to make them the same length.

4.4.4. ***Model:.*** A basic RNN model is a good baseline for sequence data.

5. Conclusion

In this research paper we tried to preset the importance and approaches to making an OCR. With its growing importance, they are becoming significantly powerful and advanced.Compared to related approaches, with increase in model's complexity it can further be designed to detect human hand writings specially from old manuscripts.

Acknowledgments

This work was made possible by all the authors and all the other sources mentioned.

References

[1] Michael D. Garris, Charles L. Wilson, , *Senior Member, IEEE, and James L. Blue*, IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 7, NO. 8, AUGUST 1998

[2] https://developpaper.com/machine-learning-sharing-knn-algorithms-and-numpy-implementation/ 1em plus

[3] Medium. 2021. A Comprehensive Guide To Convolutional Neural Networks—The ELI5 Way. [online] Available at: ¡https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53¿ [Accessed 20 January 2021]. 0.5em minus 0.4em

[4] Diva-portal.org. 2021. [online] Available at: ¡http://www.diva-portal.org/smash/get/diva2:861804/FULLTEXT01.pdf¿ [Accessed 21 January 2021]. 0.5em minus 0.4em