Annotated Bibliography

Miles Fike

September 25, 2025

References

[1] What is the model context protocol (mcp)?, 2025.

[2] Gregory Cohen, Saeed Afshar, Jonathan Tapson, and Andre Van Schaik. Emnist: Extending mnist to handwritten letters. *2017 International Joint Conference on Neural Networks (IJCNN)*, 2017.

Notably this article is peer reviewed as were all articles for the IJCNN which were later posted to the IEEE Xplore, but it is only freely accessible through Arxiv. This article for the 2017 International Joint Conference on Neural Networks was written primarily by faculty and lecturers at Western Sydney University. Professor Andre van Schaik has a Ph.D. degree in Electrical Engineering, Dr Saeed Afshar is a senior lecturer with degrees in engineering, Dr. has degrees in electrical and computer engineering and Ph.D.'s in Bioelectronics and Neuroscience and Neuromorphic engineering all with the International Centre for Neuromorphic Systems. Additionally, Professor Jonathan Tapson with a Ph.D. in Electrical Engineering, worked at the MARCS Institute for Brain, Behavior, and Development. These researchers focused on neuromorphic computing or an approach to computing inspired by the human brain, closely tied to AI. They created a dataset from the NIST dataset which used the same image structures as the MNIST dataset. This dataset is called the Extended Modified NIST or EMNIST. For context, the MNIST dataset is a set of handwritten numbers used by Ying Yu and Yuhe Tian in their article "Research Application of Computer Vision-Based Convolution Neural Network in Handwriting Recognition Technology" to develop a computer vision system. This dataset took other elements from the NIST dataset to also include alphabetical characters. Notably, this dataset contains many different writers' handwriting. This was done to provide an accessible basic dataset for testing purposes. Because of this, it was programmed with splits for testing and has data for validation. Although this dataset was meant for research, I believe that through the division of elements within lines of text it could be used as a basis for a modern handwriting computer vision system. By using this system, I would be able to interact with the datasets using any code that can interact with the very popular MNIST datasets.

[3] Peter Gyory, S. Sandra Bae, Ruhan Yang, Ellen Yi-Luen Do, and Clement Zheng. Marking material interactions with computer vision. In *Proceedings of the Conference on Human Factors in Computing Systems*, CHI '23', New York, NY, USA, 2023. Association for Computing Machinery.

This paper, written in part by the faculty of the ATLAS Institute at the University of Colorado Boulder in 2023, describes a computer vision system that can be run on a local device with a webcam. This is especially helpful as previous articles have been dependent on hardware that may not be accessible to all. This source has 5 citations showing higher engagement than the other sources investigated. The researchers in this paper investigated the creation of a Tangible User Interface system relying on computer vision rather than electronics to receive input due to the struggles novices have with devices like Arduino. They developed the beholder JavaScript library which gives users control over a device's cameras. Beholder could then use the cameras to read ArUco markers and acquire their metadata. ArUco markers resemble QR codes. After the library was designed, it was used in the projects of students. They would create systems and physical devices for which the ArUco marker could be read by Beholder, like a little arcade machine which moved the ArUco marker when a button was pressed. The system was easy to debug because visual information could indicate issues. Limitations of the system include proper lighting and space requirements for the camera. Aside from reinforcing a basic understanding of computer vision, this source is not the most relevant. However, this article does encourage consideration of elements like perspective and lightning when considering what a computer may be required to interpret.

[4] Soonhoi Ha and Eunjin Jeong. Software optimization and design methodology for low power computer vision systems. *ACM Transactions on Embedded Computing Systems*, 24(1), December 2024.

This article written by Soonhoi Ha, a professor of Computer Science and Engineering, and Eunjin Jeong, a post-doctoral researcher in the same field, in 2024 discusses computer vision systems designed both for accuracy and system constraints. It encourages the use of specialized hardware to optimize computer vision systems with deep learning. They suggest reducing redundancy in deep learning with approximate computing that does not cause significant accuracy loss. Examples in this article demonstrate systems with several convolutional layers followed by a few fully connected layers. The article goes on to discuss the results of implementing different types of quantization, quantization being the reduction of bits in representation, pruning of unnecessary parameters, reduction of large-size kernels to reduce storage requirements and computation time, and other optimization techniques. The article also strongly emphasizes when different types of hardware such as GPUs can be used to process layers. This article has been downloaded hundreds of times and continues to be downloaded frequently but it has no recorded citations on the ACM Digital Library website. The sources the article references tend to originate from scientific journals and conferences and appear credible. The article provides specific code examples for how the elements of computer vision can be programmed which would be very applicable to work in the field, and it reinforces the understanding of the basic concepts of layers in computer vision by showing examples of their implementations.

[5] Santiago Ponte Ahón, Yael Aidelman, Juan Seery, Facundo Manuel Quiroga, Franco Ronchetti, Waldo Hasperué, Matilde Iannuzzi, Romina Peralta, Mónica Lopez, Aurelio F. Bariviera, Lydia Cidale, and Roberto Gamen. ReTrOH-UNLP: Conservation of the historical observational work of the astronomical observatory of la plata with computer vision. *J. Comput. Cult. Herit.*, July 2025.

This article describes the development of the PlateUNLP application which lowers the cost and time to digitize astronomical data in plate collections. This system improves upon the collection of this data by individuals, which is an error prone and time-consuming process. The application's frontend was developed with HTML and JavaScript for a web app. A Node.js and JavaScript backend manages the storage digitized plate data storage. A deep-learning module for the detection of the object within each plate was developed with Flask and Python. PlateUNLP works by first receiving a scanned plate, then segmenting its observations, segmenting the three parts of each observation, converting the spectra from 2D to 1D, and finally the user assists in analyzing the data. The part of this process most relevant to the project is the segmentation of images. This projection created a YOLO (You Only Look Once) detection model which rather than running a classifier on regions of interests approaches segmentation as a regression problem. YOLO could be helpful for separating letters or lines of text. Additionally, the system allowed for changes in brightness, noise, vertical image flipping, rotation, and scaling to help interpret the image which could be important when using computer vision to identify letters and transcribe text if the letters are at odd angles or on problematic surfaces. The authors of this article were primarily students or researchers in Computer Science and Astronomy at The Universidad Nacional de La Plata in Buenos Aires, a university officially recognized by the Argentine Ministry of Education. It is difficult for me to thoroughly investigate each author, but all seemed to be very experienced within their fields.

[6] Mohamed Ali Souibgui, Asma Bensalah, Jialuo Chen, Alicia Fornés, and Michelle Waldispühl. A user perspective on htr methods for the automatic transcription of rare scripts: The case of codex runicus. *J. Comput. Cult. Herit.*, 15(4), March 2023.

The authors of this article include Mohamed Ali Souibgui, Asma Bensalah, Jialuo Chen, and Alicia Fornes, all of whom are researchers and engineers at the Computer Vision Center at the Autonomous University of Barcelona. They are all specialists in the field of computer vision. The journal article asserts that current Handwritten Text Recognition (HTR) methods often lack tools for end users and may struggle with rare scripts that lack labeled data for training and

fine tuning. To solve these problems, they developed and compared 4 different computer vision systems, including two cluster methods and two few shot methods. The few shot methods required slightly more user effort but resulted in higher performance, scalability, and hardware needs. Clustering involves assuming all elements are in one cluster then dividing that cluster into sub clusters until the minimum number of symbols in each cluster is reached. The best clusters are used to identify which symbol matches each cluster best. In the few shot method, each symbol is seen as a node in a graph and the similarity between each pair of symbols is measured. Comparing similarities between symbols can be scaled to compare whole lines Several parts of this paper are applicable to my wider research goals. First, the images used were preprocessed, something I will likely need to do, using binarization, which separates the foreground and background, and segmentation, which removes margins. Furthermore, segmentation is implemented to identify connected components to isolate symbols. I may need to implement these steps to transcribe lines of text. Second, the clustering and few shot methods of computer vision could provide an alternative if the standard approaches fail to work for a specific script. These could also be easier methods to use if I ever create my own data sets because they require little learning. Finally, few shot detection is good for cursive because it avoids segmentation issues.

[7] Ying Yu and Yuhe Tian. Research application of computer vision-based convolutional neural network in handwriting recognition technology. In *Proceedings of the 4th International Conference on Computer, Artificial Intelligence and Control Engineering*, CAICE '25, page 177–181, New York, NY, USA, 2025. Association for Computing Machinery.

This paper analyzes how handwriting recognition technology based on convolution neural networks work. A convolution neural network is generally used for processing grid data and is a feed forward neural network. A convolutional neural network has an input layer that takes in data like 2D images, a convolution layer, which applies kernels to an image, which extract specific features to produce an output feature map, the pooling layer which is used to compress the data and prevent overfitting by taking the averages or maximums of regions, and next, the fully connected layer is used in the last layers of the network and perform tasks like classification on each feature, and finally, the output layer returns results. Handwriting recognition technology is used to convert handwriting into a computer-processable form. This handwriting recognition technology is more widely implemented now because of AI's development and used for diverse purposes including the preservation and retrieval of ancient books. The MNIST is a data set containing handwritten digits. Pytorch was used to build a handwriting recognition model with the MNIST as its training set. The resulting model was vastly successful, but had trouble with multiple digits in succession or with incoherent strokes. The authors of this source, Ying Yu and Yuhe Tian are both associated with the College of Design and Art Shenyang Architecture University, and Yuhe Tian has researched AI in

the past, but it is difficult to determine the extent of their background in AI. The sources that the authors use were written by research scholars and faculty at other universities. Overall, the source seems reputable, but the authors are not easily researchable. This source's overview of computer vision implemented in the context of handwriting provided strong background information for any potential investigation computer vision's implementation in identifying historic ligatures which could prove challenging because the authors implied that the computer vision struggled with numbers in succession and this certainly would apply to letters as well.

[8] Xueqiang Zhang, Xiaofei Dong, Yiru Wang, Dan Zhang, and Feng Cao. A survey of multi-ai agent collaboration: Theories, technologies and applications. In *Proceedings of the 2nd Guangdong-Hong Kong-Macao Greater Bay Area International Conference on Digital Economy and Artificial Intelligence*, DEAI '25, page 1875–1881, New York, NY, USA, 2025. Association for Computing Machinery.

All authors for this journal article were researchers at the Nanjing Research Institute of Next-generation Artificial Intelligence or the China Academy of Information and Communications Technology. According to this article, AI agents convert LLMs to valuable tools with real application and problem-solving abilities. The memory of an AI agent allows it to plan how to best interact with the external environment by using function call techniques. With the multi-AI Agent collaboration system, different AI Agents can interact with each other. In the system, each AI Agent focuses on a specific domain or task and has specialized knowledge, skills, and experience in that area. Architecture can be vertical meaning that there is a lead agent or horizontal meaning all agents are equal. Agent network protocol and model context protocol are two protocols to help connect multiple agents. ANP has equal interaction between AI agents. MCP uses the model as its core and provides a common interface between all agents. AI agents may learn with technologies such as Multi agent reinforcement learning. This aims to teach optimal decision strategies through interactions with agents and their environment. Generally, multiple agent systems surpass the limitations of single agent systems, but may lack personalization, adaptability, and potentially coordination. Privacy issues are also present. This provides me with a fundamental understanding of the multiple agent interactions I wish to have controlling different computer vision systems. I may need to implement many computer vision systems to properly utilize the AI agents. This also provides very brief background on MCP.