Gestype: A Novel Solution to Ergonomically Inefficient Keyboards using Real-Time Deep Learning Based Gesture Recognition

Yaniv Briker and Richard Cheng

Abstract

In the modern workplace, physical keyboards are both an ergonomically and physiologically inefficient solution to typing, yet they are still a crucial piece of technology. Constant typing can lead to wrist problems like Repeated Strain Injuries (RSIs) or degrade lower-back strength by forcing harmful posture (Scott, 2020). Moreover, keyboards can reduce sanitation in offices as they are one of the first places that viruses begin to spread (Miller, Krennhrubec, Zuckerman, n.d.). However, keyboards have been around since 1868 and still pose significant risks due to little changes. Addressing the various health issues associated with keyboards, this project proposes an alternative method to the traditional keyboard by implementing a convolutional neural network model that recognizes hand gestures to reduce these risks. These hand gestures are determined using a binary-based counting system where each finger represents a binary state, allowing for 32 possible combinations. By using a camera, input is processed in real time through a background reduction algorithm (Figure 2); then, the processed image is input into a convolutional neural network in order to recognize the intended symbol made by the user. The symbol is customizable from user to user. The convolutional neural network predicted the correct letter 99.7% of the time. The only hardware required is a camera, making this solution versatile and easily accessible by any computer and can be run without a strong CPU or GPU.

Introduction

Deep learning, a specialized field of machine learning, is rising in popularity due to the availability of Big Data as well as its ability to do tasks quicker and superior to humans. One specific deep learning approach, convolutional neural networks (otherwise referred to as CNNs), are a main reason for the rise of deep learning. One of the causes of the popularity for CNNs is due to its ability to do human tasks, such as image recognition and classification, a useful task. Convolution neural networks are often paired with computer vision to complete image-based tasks, which is the approach presented in this project.

In a convolutional neural network, the first layer is always the convolutional layer. This layer takes an image with a fixed size and applies a kernel with a fixed size, which begins in the upper left corner of the original image. Essentially, this kernel computes multiple operations: it takes values of the original image and multiplies them by the weights set in the filter. Also, its size determines how much of an image it reads at a time. Sometimes, this kernel size may cause some information to get lost, so padding is used. Basically, it simply adds a boundary to the image with a value of zero so that it doesn't affect the computations made by a kernel. Most of the time, multiple kernels are applied and are designated to learn different things from each other. The value that the kernel computes is stored in a feature map, and one feature map is created by each kernel when it strides across an input image (Dertat, 2017). The movement of the kernel is shown in Figure 2. The main purpose of the convolutional layers is to "capture and extract the high-level features . . . such as the edges," however, "conventionally, the first convolutional layer is responsible for capturing the low-level features such as edges, color, [and] gradient orientation" (Saha, 2018). So, in summary, the number of kernels is chosen when adding a convolutional layer to the model. For example, if an input is an image with a 96x96 pixel size, and 64 kernels are applied to it, the output of the convolutional layer would be a 3 dimensional 96x96x64 (note that this value could also be 94x94x64, as it depends on the kernel size and the padding).

Then, there is a pooling layer. "After a convolution operation we usually perform pooling to reduce the dimensionality. This enables [for the reduction of] the number of parameters, which both shortens the training time and combats overfitting. Pooling layers downsample each feature map independently, reducing the height and width, keeping the depth intact" (Dertat 2017). Most of the time, max pooling is used, in which the maximum value in a pooling window is taken and placed on the pooling layer. This allows for cutting down the image shape in half (96x96 becomes 48x48).

After the model performs numerous convolutional and pooling layers, a flattening operation is set in order to convert multidimensional data made from the convolutional and pooling layers into a single row, otherwise called vectors. Then, these vectors are used in a fully connected network, in which the number of nodes represent the number of classes for classification. So, for example, in this project, the use of five fingers with two possible states creates 32 possible combinations for classes. Thus, the fully connected layer had 32 nodes, in which each one represents one class. After the data goes through these fully connected layers, it comes to the final layer where a softmax activation function acts on it. In simple terms, the model uses this activation function in order to calculate the probabilities that the image sent belongs to a particular class. Then, the class with the highest probability is chosen as the model's prediction.

Next, an activation function acts on this data. "In a neural network, the activation function is responsible for transforming the summed weighted input from the node into the activation of the node or output for that input" (Brownlee, 2019). There are many types of activation functions; however, one of the most popular ones, especially in the hidden layers of a neural network, is ReLU (rectified linear activation function). The equation of ReLU is

calculated by the equation $f(x) = max(0, x)$. Visualizing this function shows that the output for any input of less than 0 on the Cartesian plane is 0, while for any input greater than or equal to 0 the output is the input itself. Another type of activation function is called the softmax function, which is used in multiple class classification (Yang 2017). The equation for the softmax function is $\sigma(x_j) = \frac{e^{x_j}}{\sum_i e^{x_i}}$, where the input, x_j, is a vector from the last fully connected layer. This equation uses $\sum_i e^{x_i}$ to represent the sum of each normalized vector. Thus, this returns a probability that a vector (image) belongs to a class for each class; the class that the softmax function returns with the highest probability is the prediction of the model.

The usage of keyboards in both professional and personal settings can be detrimental to the sanitation of the area. For example, in hospitals, the prevalent usage of keyboards, in an area where asepsis is a necessity, can cause an outbreak which is dangerous to patients. Rutala and other UNC Health Care infection control specialists (2006) tested 25 keyboards around UNC Hospitals and found 14 types of bacteria on them, including coagulase-negative staphylococci (CoNS), which can cause dangerous bloodstream infections. Clearly, keyboards can endanger lives in an industry where keyboards are necessary to perform data analysis, order medicine, assist medical decision making, etc. Since physical keyboards are the only viable option for inputting text into electronics currently, they pose a significant risk to the health of patients because of the various forms of bacteria that preside on them. Not only are hospitals affected, but other settings with shared electronics have high levels of bacteria found on keyboards. For example, Miller, Krennhrubec, and Zuckerman (n.d.) found that shared keyboards in universities and other public areas tend to have more bacteria on them than personal computers. Furthermore, University of Arizona Researchers found there is 400 times more bacteria on a desktop compared to a toilet seat. Thus, physical keyboards have a tendency of harboring many dangerous pathogens that can survive on keyboards for prolonged periods.

Organisms	Number of Keyboards Present On
Staphylococcus aureus	1 (4%)
CoNS	20 (80%)
Bacillus spp.	22 (88%)
Micrococcus spp.	5 (20%)
Streptococcus spp.	3 (12%)
Enteric bacteria	16 (64%)
Yeasts	2 (8%)
Molds	3 (12%)

Fig. 1. Percentage evaluation of microorganisms isolated from computer keyboard surfaces (n = 25) Source: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6210060

Also, the constant repeated motions of typing can cause a variety of problems with the wrist and hand due to an excessive amount of strain on tendons. Although Repeated Strain Injuries (RSIs) can be caused by activities other than typing with a keyboard like gaming, playing an instrument, or any other repetitive tasks, typing is a common cause of RSIs. According to Scott (2020), RSIs are a type of Cumulative Trauma Disorder (CTD) where prolonged awkward motions of the hand can lead to pain throughout the upper extremities of the body. Although it might seem as if typing is a harmless movement, the repeated movements along with the long distances that fingers have to stretch in order to reach a key over hours and hours of typing can lead to damage in tendons, muscles, and nerves. Furthermore, RSIs tend to be more common when one is performing long and tedious tasks for extended periods of time, which generally match up with jobs that require typing. This makes typing with a keyboard, one of the most common sources of pain in the arms, much more dangerous than the harmless task it seems like. One factor that promotes the development of RSIs is poor posture, which is generally another health problem attributed to office jobs typically involving keyboards.

As previously mentioned, keyboards can also cause bad posture, which contributes to a variety of other problems. For example, bad posture accelerates the development of RSIs due to the bad positioning of the wrists that can cause more wear on essential parts of the wrists. The cause of the bad posture is the ergonomically inefficient position that typing forces the human body into. For example, Cornell University (n.d.) found that having a desk-top keyboard makes it more difficult to place the wrists in a neutral position along with compressing median and ulnar nerves, reducing blood flow to the fingers. Thus, working for anywhere past 3-4 hours in this position will result in muscle fatigue, making the problem with posture even worse. Keyboards can even cause problems with back and neck pain when used extensively. Burr (2019) states that when sitting down, the lower back experiences 90% more pressure than when standing, thus causing more tension on the neck and back when sitting down. Even further, sitting down for prolonged periods of time results in weakened glutes and core, causing a shift in the pelvis called an Anterior Pelvic Tilt (APT), eventually leading to lower back pain, poor movement mechanics, and reciprocal inhibition. Thus, sitting down forces the body into a hunched position, weakening muscles and causing pain throughout many muscles that support the posture.

Methodology

In order to create a standardized set of gestures, a system needs to be used in order to make sure all gestures are in some sort of a pattern. Taking an idea from encryption using binary operators, fingers can act as a boolean with binary states. Similarly to a study done by Deepak and Parveen (2017), binary numbers were used to represent letters of the alphabet to be used to encrypt strings by writing the ASCII number of a symbol in binary form. A similar version of this could be adapted in order to represent at least 26 symbols as five digits can represent 32 symbols in the binary system. Using a numbering system when dealing with a language helps classification and keeping such processes organized and well structured. This way, each symbol can easily be customizable as numbers can be assigned to strings, allowing for a myriad of possibilities from a simple system with a finger either pointing up, or not pointing up. Simply, when a finger is pointing up, it represents

a digit of "1", and when a finger is held down, it represents a digit of "0". This offers a simple solution to having to memorize gestures as it reduces the complexity and variance of each gesture to simple states for each digit.

Problem

Keyboards lead to many issues with the spread of viruses and bacteria, as well as damage to posture and the development of Repeated Strain Injuries. To explain, many types of bacteria are known to be able to reside on a keyboard for prolonged periods, some up to 24 hours (UNC Health Care), causing major health problems. Also, keyboards force users into bad posture, putting excess strain on the lower back and neck, causing pain and reducing strength in necessary muscles. Finally, the repeated movement of typing with awkward stretching on the keyboard can lead to strain on the nerves, tendons, and muscles in the upper extremities.

Objective

Propose an easily accessible method of inputting text into a computational system using only input from a common camera.

Dataset

About 2,000 images were collected for each gesture.

Features applied:

- Background Subtraction (MOG2 subtraction)
- Convert to Grayscale
- Gaussian Blur
- Binary Thresholding
- Finding Hand Contour => Hand Segmentation
- Convex Hull & Convexity Defects

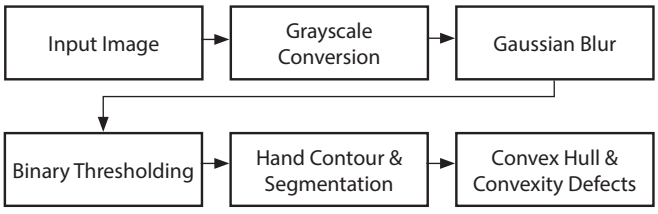

Fig. 3. Diagram for Image Processing

Gestures Available for Signaling

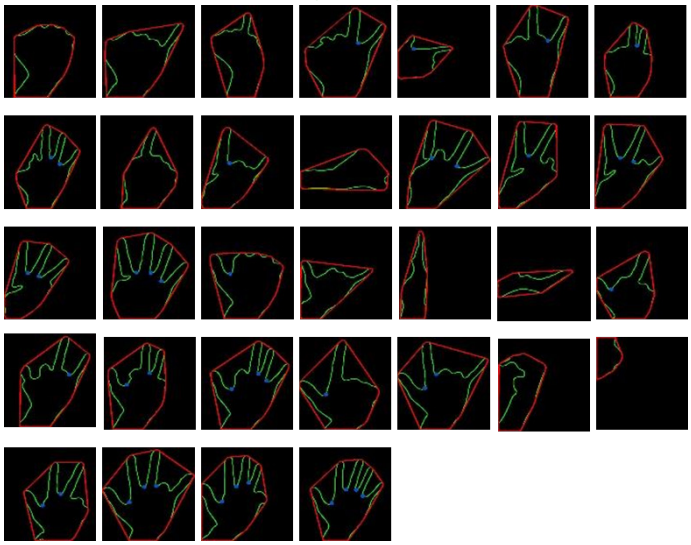

Fig. 4. Current Gestures for Gestype. These gestures can be set to any alphanumeric value on the keyboard, allowing for full user customizability.

Real Time Application

- Users can choose their own interval on which the application takes a frame in to predict on.
- The application asks for users to click "B" to set the background in order for the image processing algorithm to work.
- Clicking "R" allows for the background to be reset, allowing for the user to refresh the background the image processing algorithm uses.
- Any computer, regardless of CPU or GPU, can run the application, allowing for world-wide accessibility and usability.

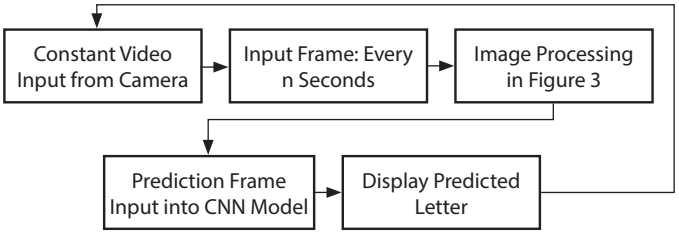

Fig. 5. Diagram for Real-Time Prediction

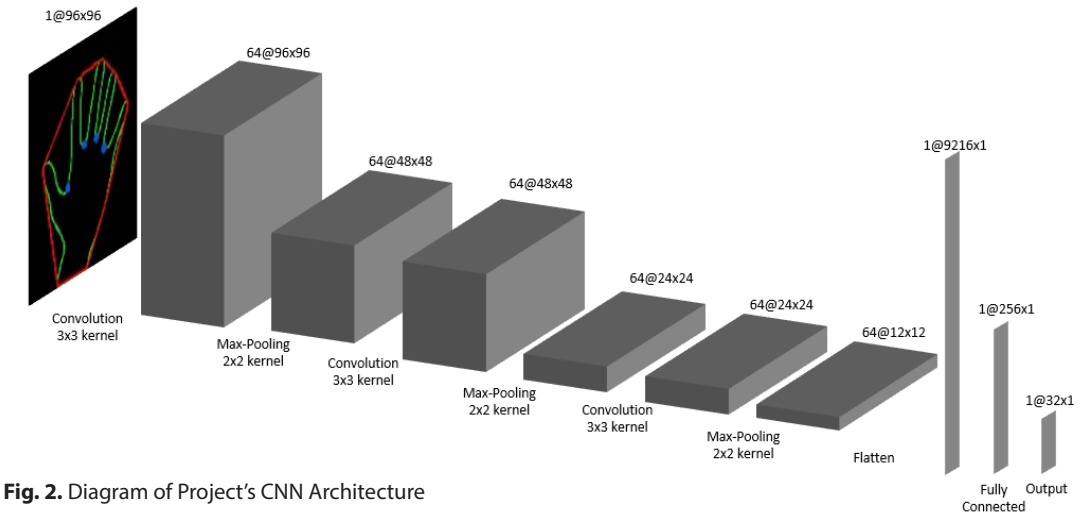

Fig. 2. Diagram of Project's CNN Architecture

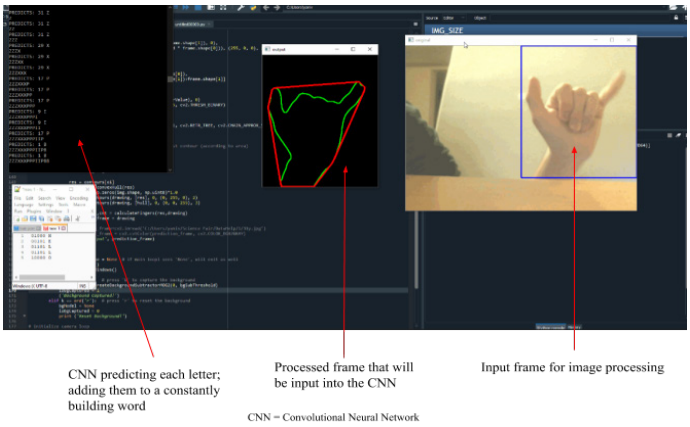

Fig. 6. Display of Functioning Gestype

Model Statistics

Training Accuracy	Training Loss	Validation Accuracy	Validation Loss
99.97%	0.15%	99.89%	0.66%

Precision $P = \frac{tp}{tp + fp}$	**Recall** $R = \frac{tp}{tp + fn}$	**F_1 Score** $F_1 = 2 \times \frac{P \times R}{P + R}$
99.84%	99.75%	99.79%

tp=true positive; fp=false positive; fn=false negative.

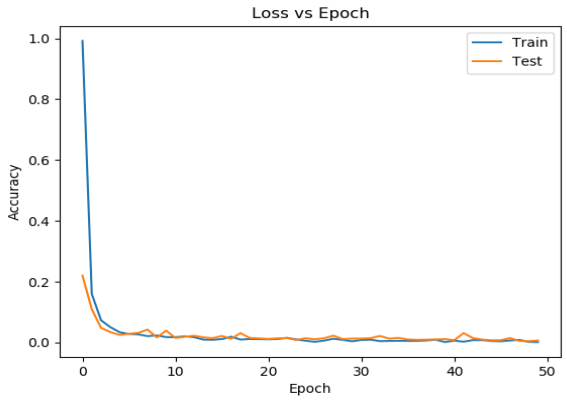

Fig. 7. Graph 1: The model's accuracy increases as the epochs increase because it learns features from the images and adjusts itself to predict correctly on new input. The model reached 99.97% accuracy by the end of 50 epochs of training.

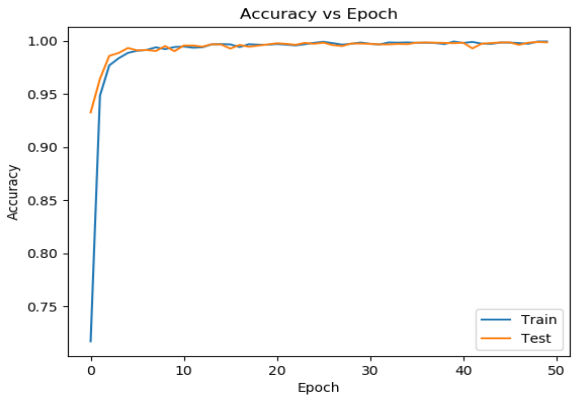

Fig. 8. Graph 2: The model's error decreases as the epochs increase due to the fact that it is learning and increasing its accuracy. The error is virtually zero by the end of the 50 epochs training.

Conclusion

Gestype is an approach to typing that reduces the transfer of bacteria. Standard keyboards have augmented the current state of technology; however, they have also resulted in the rapid and unsanitary spread of germs and viruses along with the deterioration of posture and health. Gestype is a novel method that uses a camera in order to recognize preset gestures to potentially replace the conventional method of keyboard input. By adding more features such as customizable gestures and sentiment analysis, Gestype is improving, showing the world a message about the importance of having a safe way to type. This suggests that Gestype could have a widespread use in the modern day as it offers many benefits with little drawbacks. Long term effects of Gestype have yet to be tested. In order to evaluate the effect on RSIs, more testing has to be done with various gestures and how they can possibly affect the health of muscles, nerves, and tendons.

Gestype removes the need for a physical keyboard, and thus removes the risks involved with a physical keyboard. The convolutional neural network had a very high validation accuracy (99.89%) and training accuracy (99.97%). This indicates that Gestype is a viable tool that can be used while being as accurate, but healthier, than a physical keyboard. Although the project was mostly successful, there is definitely room for improvement. One source of error presents itself in image processing: shadows interrupt the algorithm used to segment the hand, which messes up the output of the algorithm. This could be solved by thinking of a new algorithm for image processing. Another potential problem was that data was only collected from two users; however, this is easily resolvable with new added features, such as user-customizable gestures and gesture sharing between multiple users.

This idea can be extended to other complicated gesture recognition and technology interaction systems eventually allowing seamless integration of technology into everyday life. The central idea of Gestype is to use simple gestures that can be recognized in real time and implemented to elicit a simple response from technology. This could also be implemented in frameworks such as augmented reality (AR) systems.

One of the drawbacks to Gestype is the requirement to memorize a completely new system of gestures in order to type. The new system was optimized to be as simple as possible, but still required a completely new mechanism due to no easy representations of the English lexicon. Another drawback is that each person will have different mobilities for each gesture, requiring a newly trained model for some users. This adds on an additional requirement for CPU or GPU caused by the limited set of training data collected. This creates a tradeoff between accuracy and difficulty of usage as requiring the training a new model might not be accessible to all computers. For a more complicated system, the rate at which accuracy increases in relation to difficulty is uncertain.

Future Research

Add Natural Language Processing (NLP) along with emotion detection in order to autocorrect text after input using gestures. Also, potentially allow for more customization by letting users add gestures to a personal database and train their own gestures, as well

as import other users' gestures. Future features may also include dual hand typing, in which there are two cameras recording at the same time, each one focused on each of the person's hands.

References

Brownlee, J. (2019, August 6). A Gentle Introduction to the Rectified Linear Unit (ReLU). Retrieved from https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks/

Burr, R. J. (2018, April 3). How Sitting Causes Back Pain. Retrieved from https://www.startstanding.org/sitting-back-pain/

Deepak, & Parveen. (n.d.). Modern Encryption and Decryption Algorithm based on ASCII Value and Binary Operations. Retrieved from https://pdfs.semanticscholar.org/d718/402660e0d9d8bca1f5cd37ecd18306ae4d90.pdf

Dertat, A. (2017, November 13). Applied Deep Learning - Part 4: Convolutional Neural Networks. Retrieved from https://towardsdatascience.com/applied-deep-learning-part-4-convolutional-neural-networks-584bc134c1e2

Ideal typing posture: Negative slope keyboard support. (n.d.). Retrieved from http://ergo.human.cornell.edu/AHTutorials/typingposture.html

Koscova, J., Hurnikova, Z., & Pistl, J. (2018, October 12). Degree of Bacterial Contamination of Mobile Phone and Computer Keyboard Surfaces and Efficacy of Disinfection with Chlorhexidine Digluconate and Triclosan to Its Reduction. Retrieved from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6210060/

Miller, S., Krennhrubec, K., & Zuckerman, D. (2017, March 24). Are There More Bacteria on Computer Keyboards Than Toilet Seats? Retrieved from http://www.center4research.org/bacteria-computer-keyboards-toilet-seats/

Repetitive Strain Injury. (n.d.). Retrieved from https://web.eecs.umich.edu/~cscott/rsi.html

Rutala, White, M. S., Gergen, M. F., & Weber, D. J. (2008, February 14). Computer keyboards in health-care settings should be disinfected daily, UNC Health Care study concludes. Retrieved from https://healthtalk.unchealthcare.org/keyboards/

Saha, S. (2018, December 17). A Comprehensive Guide to Convolutional Neural Networks-the ELI5 way. Retrieved from https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53