Early Detection of Acromegaly Using a Novel Convolutional Neural Network

By Anish Leekkala, Ukash Nakarmi

Bentonville High School / University of Arkansas

Abstract

Acromegaly occurs when the pituitary gland produces too much somatropin, causing the liver to release excessive amounts of IGF-1, leading to the abnormal growth of the hands, feet, and face. Acromegaly is difficult to diagnose and can lead to serious, sometimes even life-threatening, health problems such as Type II diabetes and heart disease. The early detection of Acromegaly reduces potential health complications and the risk of death. Deep Learning assisted early detection has now been proven feasible according to latest research and the prevalent success of Transfer Learning provides a potential path of non-computationally intensive detection. In this study, a dataset containing roughly 20 images were used to train a Convolutional Neural Network with Transfer learning that utilized ResNet-18 to mitigate the low dataset size. Firstly, Acromegaly and Non-Acromegaly images were placed into separate datasets and were further separated in a 70/30 training-validation split. This was run through the model, achieving a 65.62% validation accuracy over 25 epochs. This was paired with high training-validation loss values, 0.7/1.5 respectively, past epoch 25. To improve these losses, pairs of the same person were used to mitigate data imbalance within the datasets occurring from multiple same patient images. The datasets were comprised of Acromegaly/Non-Acromegaly (A/N) pairs and Non-Acromegaly/Non-Acromegaly (N/N) pairs and each pair was fed through a custom data loader to then be fed through two Resnet-18 models, which were able to train on the differences between normal (N/N) and abnormal (A/N) growth. This led to a 9.9% validation increase as well as training/validation loss values of (0.6/0.65) by epoch 25. This proved that non-computationally intensive detection of Acromegaly was possible with limited data, and a lightweight model could be distributed and used to assist doctors/researchers on whether a patient needs to test for Acromegaly or not, saving lives.

Keywords

Acromegaly, Convolutional Neural Network, Transfer Learning, Early Detection, Student Paper

Motivation and Approach

Acromegaly is rare and often difficult to diagnose (cases were between 71.0 and 87.0 people per million people) (Broder et al. [1]), which causes some patients to suffer from the condition for years before receiving an accurate diagnosis (NORD [2]). Once the problem is identified, state-of-the-art techniques are available to treat Acromegaly, so the crux of the problem is diagnosing the disease. Therefore, I hope to detect Acromegaly early before long-term effects such as: Heart Disease, High Blood Pressure, High Cholesterol, Type 2 Diabetes, Enlargement of the Thyroid Gland, Sleep Apnea, and Carpal Tunnel Syndrome (Mayo Clinic [3], NIDDK [4]). Utilizing

Deep Learning will provide the solution to early detection by highlighting patterns and changes that are too gradual for humans to notice until after long periods of time.

My approach is to search for the most common Acromegaly features that can be tracked by a Deep Learning Model: Abnormally Large Hands and Feet, Pronounced Facial Features, and an Enlarged Tongue (Duan et al. [5]). I gathered data that focuses on these features from participants by reaching out to Acromegaly Communities and hospitals to collect as much data as possible. This data comprised of two primary categories, non-Acromegaly patients (this set will comprise the images of Acromegaly patients before they have the disease) and Acromegaly patients (the patients after they have been diagnosed with the disease). I then trained a CNN model on this data (Kong et al. [6], Kong et al. [7]). After developing my model, I plan to push the model to a cloud server. I will then build a mobile application, designed in React Native and Java, which will allow researchers to be able to upload images of test patients, and the model will then be trained on those images and give a decision that will help researchers determine the likelihood of Acromegaly in a patient. Because Acromegaly is a slowly occurring disease, the researchers will only need to upload new patient images once every few weeks or they can also train the model on past images of the same patient (such as a dataset of the patients' images) to create a diagnosis. My goal is for this mobile app to be released to the app store, but to release it only for private distribution. I will provide the download link specifically to select researchers and doctors, as only people with licensed medical degrees can make a diagnosis, so by releasing it to only medical researchers / doctors, they will be able to use my app as a tool to help diagnose patients with Acromegaly and implement it in their own research.

Logistics and Organization

The data for this model comes from patients who have had Acromegaly before. Because of Acromegaly's rarity, finding patients is extremely difficult, so the pictures could not be normalized in such a way that every photo is taken from the same camera or in the same format to minimize accuracy loss. I received the data by reaching out to hospitals and Acromegaly communities, with a director of an Acromegaly Community giving me permission to reach out to patients for their voluntary information.

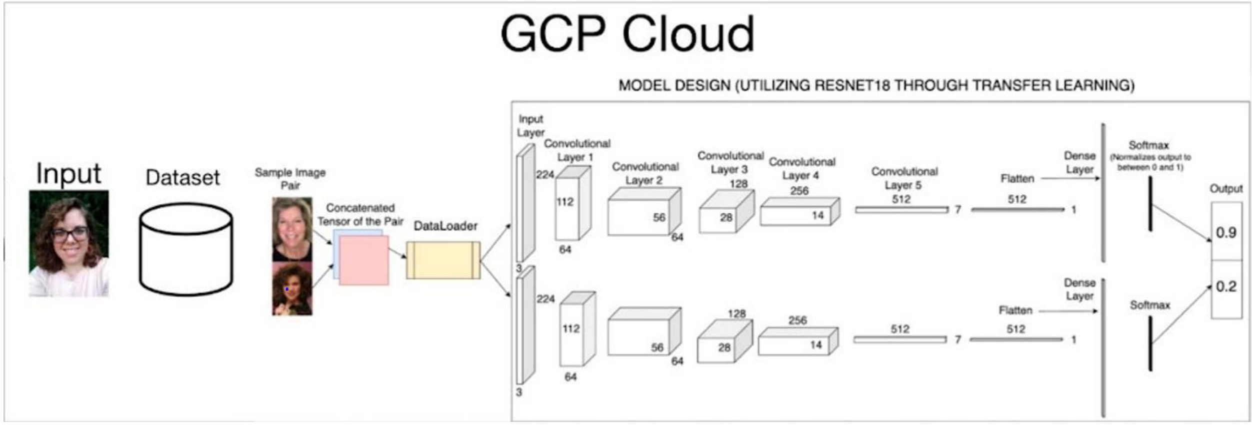
To receive more data to train the model effectively and efficiently in the future, I will use a GAN, a Generative Adversarial Network to create more training images (Goodfellow et al. [8]). A GAN works by utilizing two models, the Generator Model, and the Discriminator Model. The discriminator is trained to recognize and differentiate between Acromegaly and non-Acromegaly, which my current model can be used for. The model will be trained on the "domain dataset" containing real Acromegaly images. When the Discriminator has a high enough success rate, the Generator Model will create new "fake" images using photos of people from the dataset to pass to the Discriminator Model. The Generator will try to create a picture in which the Discriminator Model can identify Acromegaly, and the Generator Model will train itself and create thousands of pictures to feed to the Discriminator (Goodfellow et al. [8], Aggarwal et al. [9]).

To train the CNN model, the patient data was first put into a 70/30 training and validation split. In each of those folders, I further split the model into Acromegaly and non-Acromegaly, making sure to minimize the occurrences of the same person showing in the training and the validation set (Koch et al [10]). This method ensured the model did not overfit the data, as the same picture showing up twice would make the model immediately classify pictures of the same person as “Acromegaly” without properly getting the chance to “understand” or assign the necessary weights for the model. However, it was essential that the same person was in both the Acromegaly and the non-Acromegaly datasets, as it is beneficial for the model to learn the difference between when a person has Acromegaly or not. For the model itself, I used Facial Feature Extraction (FFE), which allowed the model to recognize the key landmarks of the face and more easily tell the physical differences of a person with Acromegaly, especially by looking for shrunken eyes and a pronounced nose. However, to do this, the images had to be effectively cropped, so the FFE can extract the key landmarks of the face. In the future, I plan to implement automatic facial detection using a Cascade CNN model (Zhang et al. [11]).

The model is a pre-trained CNN, already having the necessary weights and neural layers needed to accurately classify a face. A picture from the dataset is fed into the CNN, which is then broken apart into a grid of squares, each containing parts of the image, which varies depending on the size of the image, and this is then converted into a tensor (Koch et al. [10]). To make the model more accurate, the squares can be flipped, reversed, dilated, etc. through a Data Loader so that the model can learn to recognize Acromegaly in all forms to increase the validation accuracy. The tensor is then broken apart and fed through the neural network and through the pre-trained layers of the network which already have all the previous weights from the pre-trained model, except for the last layer, which is completely randomized for familiarization with the model. This model was then run through 24 epochs, and the result was used to display the model accuracy by taking a random set of the training images from the last epoch and displaying the model’s diagnosis. The first iteration of this model struggled to deal with multiple images of the same person and overfit the data. This caused the training accuracy to be up to 20% greater than the validation accuracy while the training loss was less than the validation loss. I saw the potential, however, as I still managed to get 86.49% highest training accuracy and a 65.62% highest validation accuracy using the ResNet pretrained model (Kappeler et al. [12], He et al. [13]).

To fix this issue, I placed two images of the same person together into the same tensor. Instead of the tensor having the dimensions of the images’ width, height, and having a 3 for the third dimension to account for the RGB channels in the pictures, this solution changed the images’ third dimension to 6 because the images were stacked on top of each other (Kappeler et al. [12], Nandy et al. [14]). Instead of having the Acromegaly and the Non-Acromegaly Dataset, I am using a 0-0 dataset and a 0-1 dataset with 0 meaning no Acromegaly and 1 meaning Acromegaly is present. This means that in the 0-0 dataset I am putting two pictures of the same person before they had Acromegaly and in the 0-1 dataset, I am putting the person before Acromegaly and the person after Acromegaly together. I am not using a 1-1 dataset because this model is being trained on the development of Acromegaly, so it needs to identify the difference between normal development and Acromegaly-induced development. I achieved a 9.9% increase in the validation accuracy here, going from 65.62% to 75.53% percent accuracy. I also had a 3.6% increase in my training accuracy, going from 86.42% to 90%, while still having this new model being pre-trained on the ResNet model (Kong et al. [7]).

Photo Credits to Ms. Elizabeth Porch and Ms. Wendy Biggers



The above diagram shows the sample image pair with the person on top having Acromegaly and the same person on the bottom without Acromegaly. The output on the right indicates that the top image has a high chance of having Acromegaly (if the SoftMax throws a value that is less than 0.5, then it shows no Acromegaly, otherwise it shows Acromegaly) and the bottom has a high chance of not having Acromegaly (Kong et al. [6], Koch et al. [10]).

This model in its current state can only look at pictures of people who have or do not have Acromegaly and tell the likelihood that they have Acromegaly. When the app is released to researchers, I will take a slightly different approach to the model. Since the model will already be trained to a reasonably high accuracy (at least 95%), the model can be trained on a patient dataset of the same person and be used to classify the chances of Acromegaly for that patient. When the model sees a repeating occurrence of Acromegaly (5-10+ recent photos in which Acromegaly is identified), the app will tell the researchers that this patient has a chance for Acromegaly.

This solution takes the best features from existing options that utilize a CNN model but adds the unique advantages of portability and increases in accuracy by utilizing 5-10 *mostly consecutive* recent photos of the person to reduce the chance of false detection from the single snapshot method. Even at 5 consecutive photos with a 5% chance of misdiagnosis, there is a 5% x 5% x 5% x 5% x 5% or 0.00003125% chance of a misdiagnosis, which ensures accurate judgement by the AI. Additionally, because my model will be pre trained and on a cloud server, I will only have to re-train the model whenever the server reboots.

Risks and Their Solutions

A risk that could be associated with the project is the images that the researchers feed to the model may not be enough to predict Acromegaly in a person. Because the accuracy of the data is dependent on physical features, sometimes the model might not get good enough pictures to show a clear view of the features (Shin et al. [15]). The solution here is to rely on the other features for detection, or the model can potentially reject those images and ask the researcher for different images (He et al. [13], Xin et al. [16]).

One of the most important risks with this model is data collection. As stated in the data section of the model, collecting data for Acromegaly is difficult due to the rarity of this disease. While I plan to rectify this issue using a GAN, the GAN still needs sample data to train on. I will continue to reach out to more hospitals and other Acromegaly Communities to get more data so that I can begin training the GAN (Goodfellow et al. [8]).

Current Progress

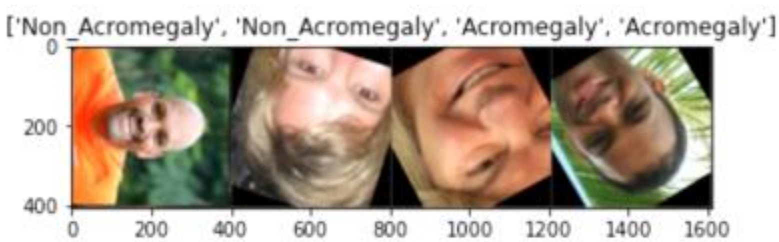
Data

As mentioned in the data and model section of this paper, I have already collected 20 pairs of images from a variety of donors from a multitude of age groups to ensure model accuracy and utilize my limited dataset to train the model. I currently have strong ties with multiple Acromegaly leaders in their communities, which has allowed me to steadily collect data and photos to keep increasing the accuracy of the model.

Model Development

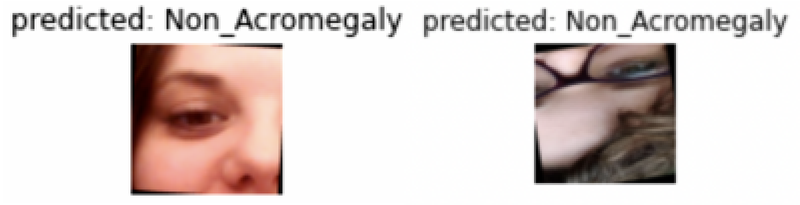
I have already built and tested the prototype version of the CNN model, coming up with an 86.49% highest training accuracy and a 65.62% highest validation accuracy using the ResNet pretrained model. However, I am currently testing the model that takes two images of the same person and puts them together to create 0-0 pairs and 0-1 pairs. Utilizing the difference between the faces of the same person means that the model is less likely to overfit, and the model clearly learns the differences between the growths of Acromegaly and Non-Acromegaly. Instead of training on all the faces between the groups of Acromegaly and Non-Acromegaly at the same time the model looks at pairs of images in batches, so the model can more easily compare the various features of the faces together and see the differences between them (Nandy et al [14], Shin et al [15]).

Photo Credits to Mr. Chetan Malik, Ms. Wendy Biggers



This image represents a batch of 4 images from the data preprocessing part of my model, which my Data Loader takes care of. The pairs of images mentioned in the data portion can also be seen here, with the 1st and 4th images being the same person and the second and third being the same person (the model randomly rotates the images when I display it). Because the model only accepts the same image, the photos are resized to a 500x500 resolution, and the model gets a 224x224 crop from it (Duan et al [5]).

Photo Credit to Ms. Elizabeth Porch

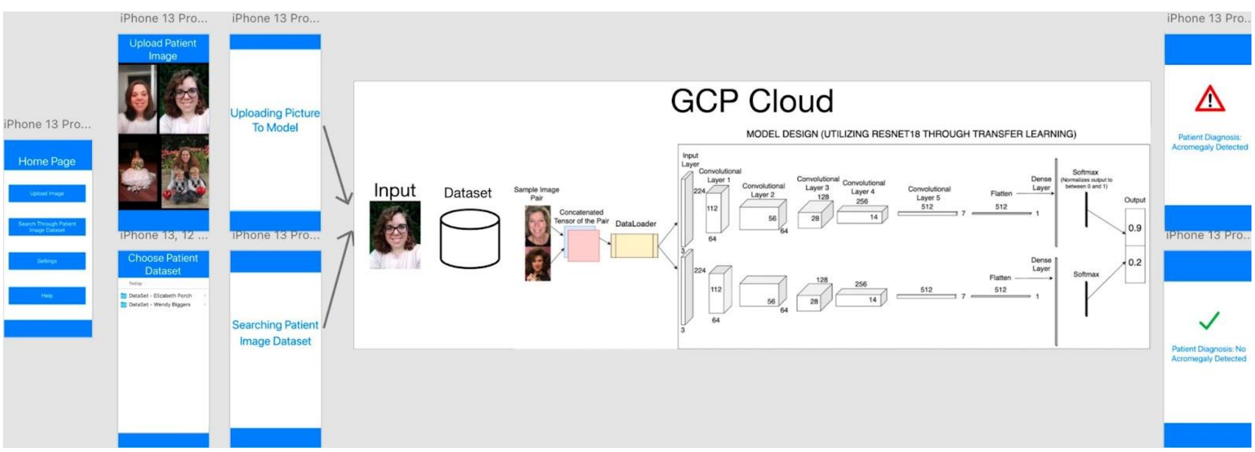


After the data goes through the Data Loader and the CNN Model, the program displays 5 random training images and presents its predictions above each of the images. These images above were taken from a 0-0 pair and were both correctly identified as not having Acromegaly.

App Building

As of right now, I have planned out much of the app's UI, but because I want the app to be given to researchers to help them check for Acromegaly in patients, the design is going to be very minimal and will just consist of two paths. The first will allow the researchers to upload new patient images which will get added to the existing patient dataset, and the other allows the researchers to train on that dataset to see if there is a chance for Acromegaly as illustrated by the picture below.

Photo Credits to Ms. Elizabeth Porch and Ms. Wendy Biggers



Personal Interest

My interest in taking on this project stemmed from my dad who was affected by Acromegaly. When I discussed the condition with him, he mentioned that he had the disease for ten years and he was not aware of it until recently. His brother, a doctor, had not seen him for the last ten years visited their hometown and diagnosed him with Acromegaly. This has led me to realize that with Deep Learning, this disease could have been diagnosed a lot sooner, and that there are thousands silently suffering from this disease that I could help. My recent research on Deep Learning inspired me to take on the challenge of early diagnosis, and I have researched numerous papers to understand Computer Vision and Facial Recognition to take on this task. I also have years of app development experience, so I was uniquely poised to create a mobile app that can achieve quick, easy diagnosis of Acromegaly so that I can help all those get the treatment they need to prevent Acromegaly from permanently harming their lives.

References

[1] M. S. Broder, E. Chang, D. Cherepanov, M. P. Neary, and W. H. Ludlam, "INCIDENCE AND PREVALENCE OF ACROMEGALY IN THE UNITED STATES: A CLAIMS-BASED ANALYSIS," *Endocrine practice: official journal of the American College of Endocrinology and the American Association of Clinical Endocrinologists*, vol. 22, no. 11, pp. 1327–1335, 2016, doi: 10.4158/EP161397.OR.

[2] "Acromegaly," NORD (National Organization for Rare Disorders).

[3] Mayo Clinic, "Acromegaly - Symptoms and causes," Mayo Clinic, 2019.

[4] "Acromegaly | NIDDK," National Institute of Diabetes and Digestive and Kidney Diseases, May 02, 2019.

[5] C. Duan *et al.*, "Automatic Detection for Acromegaly Using Hand Photographs: A Deep-Learning Approach," *IEEE Access*, vol. 9, pp. 2846–2853, 2021, doi: 10.1109/access.2020.3046878.

[6] Y. Kong et al., "Constructing an automatic diagnosis and severity-classification model for acromegaly using facial photographs by deep learning," Journal of Hematology & Oncology, vol. 13, no. 1, Jul. 2020, doi: 10.1186/s13045-020-00925-y.

[7] X. Kong, S. Gong, L. Su, N. Howard, and Y. Kong, "Automatic Detection of Acromegaly From Facial Photographs Using Machine Learning Methods," EBioMedicine, vol. 27, pp. 94–102, Jan. 2018, doi: 10.1016/j.ebiom.2017.12.015.

[8] I. J. Goodfellow *et al.*, "Generative Adversarial Networks," 2014, doi: 10.48550/arXiv.1406.2661.

[9] Aggarwal, M. Mittal, and G. Battineni, "Generative adversarial network: An overview of theory and applications," *International Journal of Information Management Data Insights*, p. 100004, Jan. 2021, doi: 10.1016/j.jjimei.2020.100004.

[10] G. Koch, R. Zemel, & R. Salakhutdinov. Siamese Neural Networks for One-shot Image Recognition. Department of Computer Science, ICML deep learning workshop, vol. 2. 2015

[11] J. Zhang, Z. Wang, Y. Zheng, and G. Zhang, "Cascaded Convolutional Neural Network for Image Super-Resolution," Advances in Artificial Intelligence and Security, pp. 361–373, 2021, doi: 10.1007/978-3-030-78615-1_32.

[12] A. Kappeler, S. Yoo, Q. Dai, and A. K. Katsaggelos, “Video Super-Resolution With Convolutional Neural Networks,” IEEE Transactions on Computational Imaging, vol. 2, no. 2, pp. 109–122, Jun. 2016, doi: 10.1109/tci.2016.2532323.

[13] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” arXiv.org, 2015. doi: 10.48550/arXiv.1512.03385

[14] A. Nandy, S. Haldar, S. Banerjee, and S. Mitra, “A Survey on Applications of Siamese Neural Networks in Computer Vision,” 2020 International Conference for Emerging Technology (INCET), Jun. 2020, doi: 10.1109/incet49848.2020.9153977.

[15] H.-C. Shin et al., “Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning,” IEEE Transactions on Medical Imaging, vol. 35, no. 5, pp. 1285–1298, May 2016, doi: 10.1109/tmi.2016.2528162.

[16] M. Xin and Y. Wang, “Research on image classification model based on deep convolution neural network,” EURASIP Journal on Image and Video Processing, vol. 2019, no. 1, Feb. 2019, doi: 10.1186/s13640-019-0417-8.

Anish Leekkala

Anish Leekkala is a senior from Bentonville High School, and is currently certified in Python, Java, and React Native, and has expertise in Artificial Intelligence / Machine Learning. Anish is currently conducting research on Acromegaly and MRI Motion Artifacts at the University of Arkansas under Dr. Nakarmi, as well as writing a Systematic Review on Subdural Hematoma through the Saint Michael’s Hospital in Canada under Dr. Cusimano. Anish is also a Software Developer Intern at Vertalo.

Ukash Nakarmi

Dr. Ukash Nakarmi is currently working as an Assistant Professor in the Department of Computer Science / Computer Engineering at the University of Arkansas. He has received his PHD from the University of New York in Buffalo and is a postdoctoral fellow at Stanford University. He has been cited in over 200 papers and he is currently interested in research relating to Machine Learning and Artificial Intelligence, especially regarding computational / medical imaging and healthcare.